PMR 5237

Modelagem e Design de Sistemas Discretos em Redes de Petri

Aula 6: Redes de Alto Nível e Redes Coloridas

Prof. José Reinaldo Silva
reinaldo@usp.br

Applications

Industry Applications

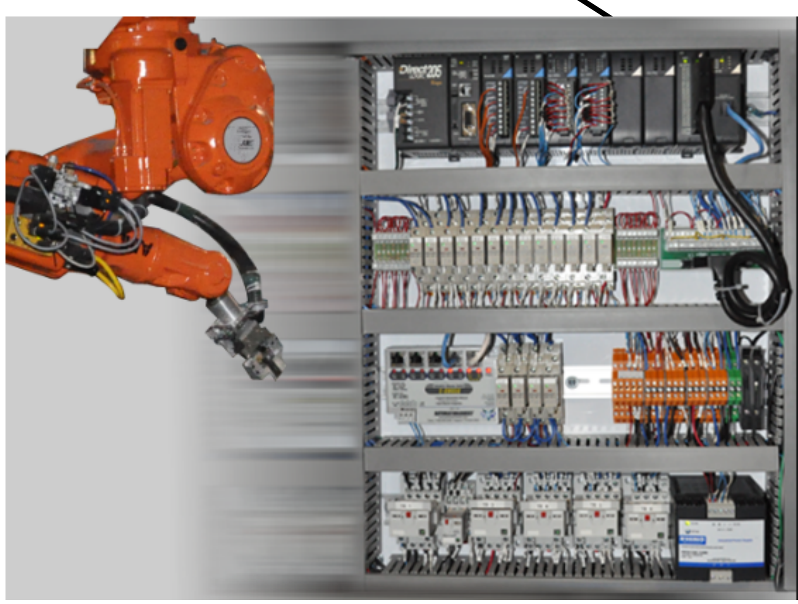
manufacturing

PLC's

process industry

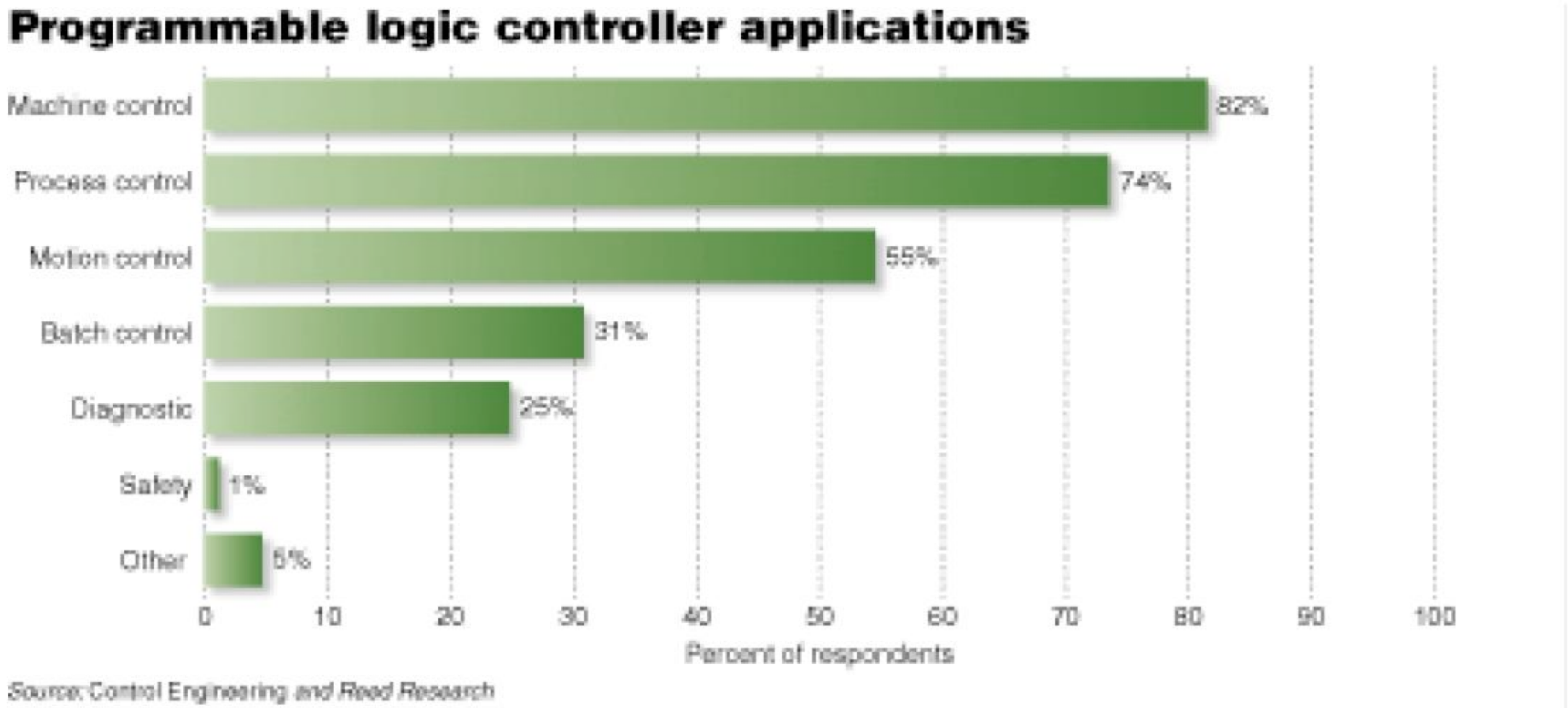
SDCD's

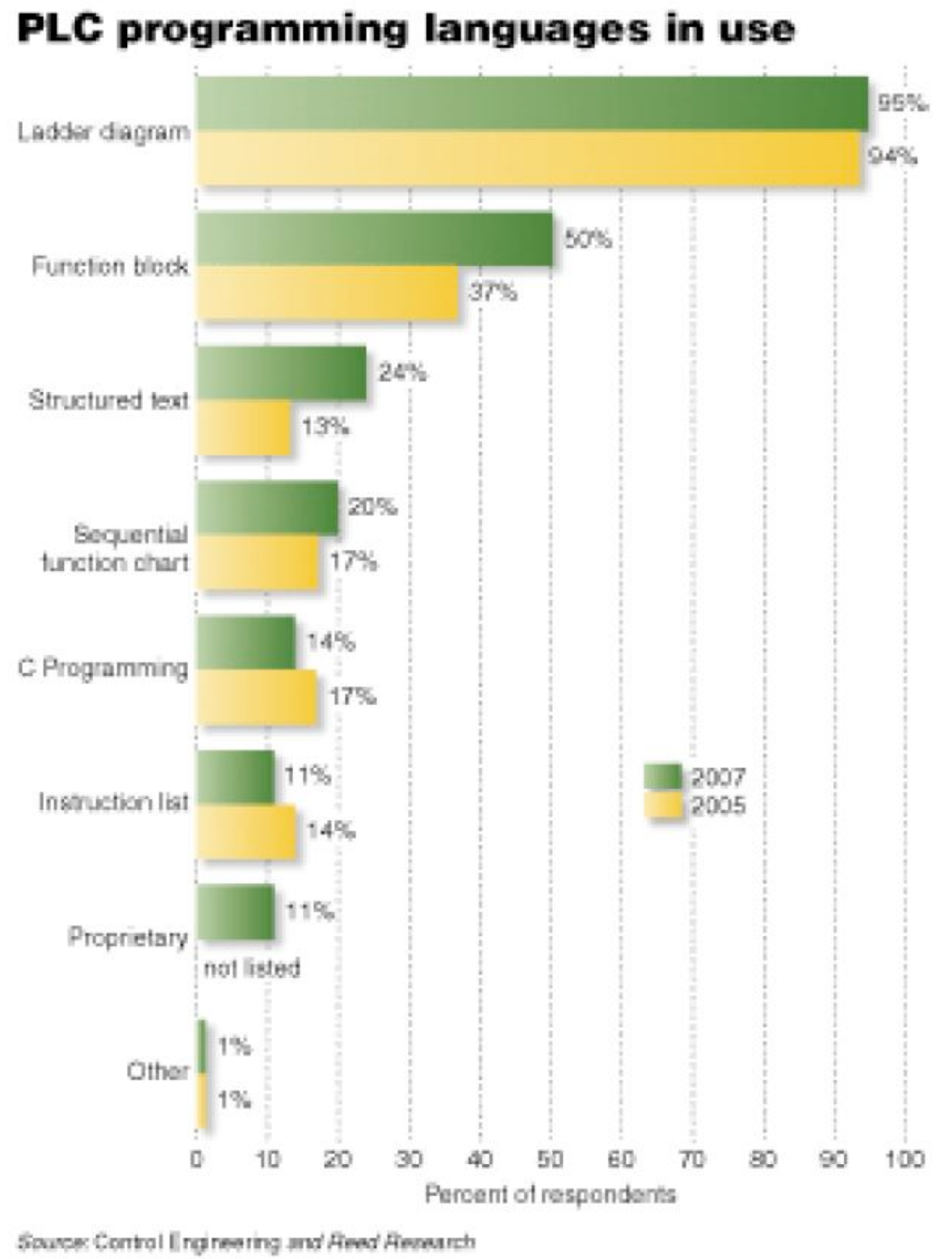
(Abstract) Models



Aplicações das RdP em processos industriais

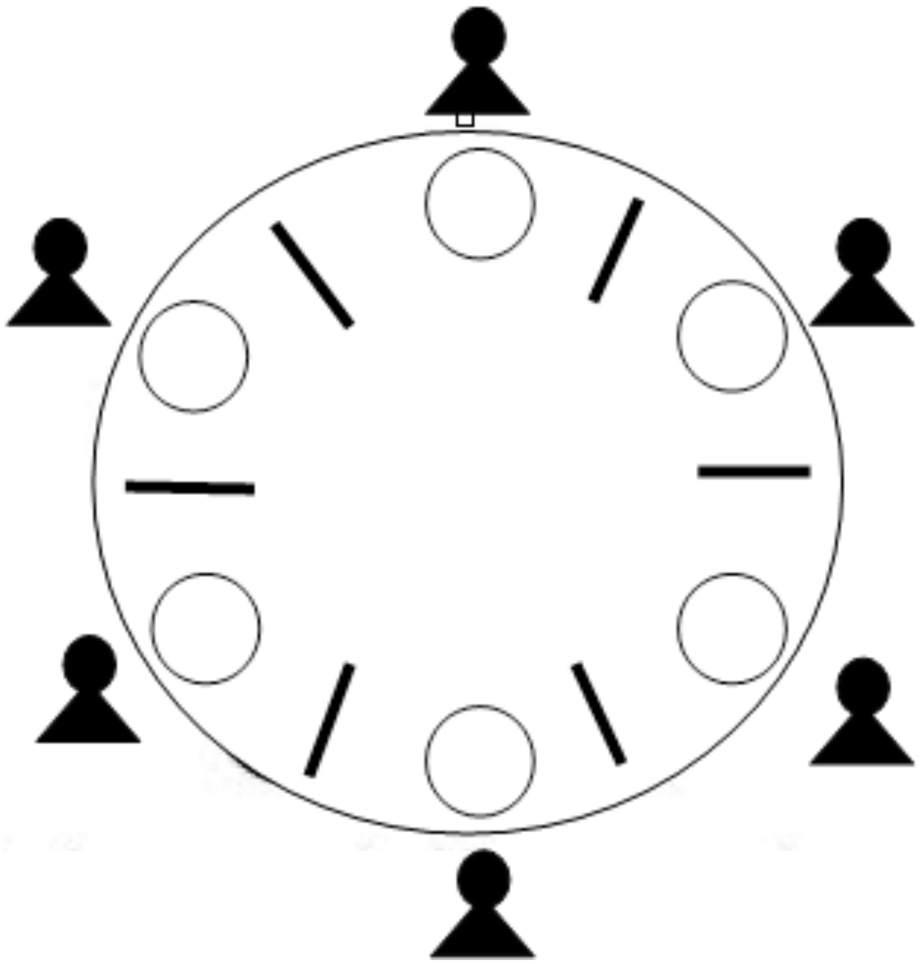
Dick Johnson, Control Engineering -- Control Engineering, 12/1/2007




PLC programming languages in use
Ladder diagram
95%
94%
Function block
50%
37%
Structured text
24%
13%
Sequential function chart
20%
17%
C Programming
14%
17%
Instruction list
11%
14%
Proprietary
11%
not listed
Other
1%
1%
2007
2005
0
10
20
30
40
50
60
70
80
90
100
Percent of respondents
Source: Control Engineering and Reed Research

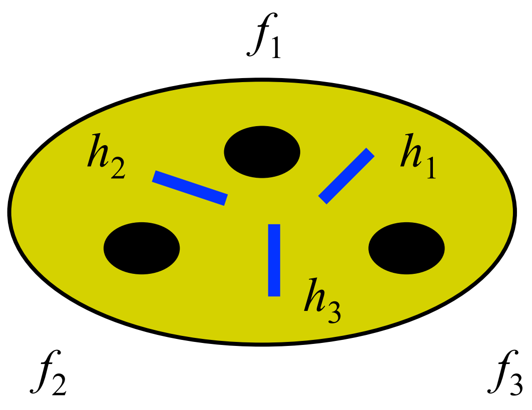
The dinning philosophers problem

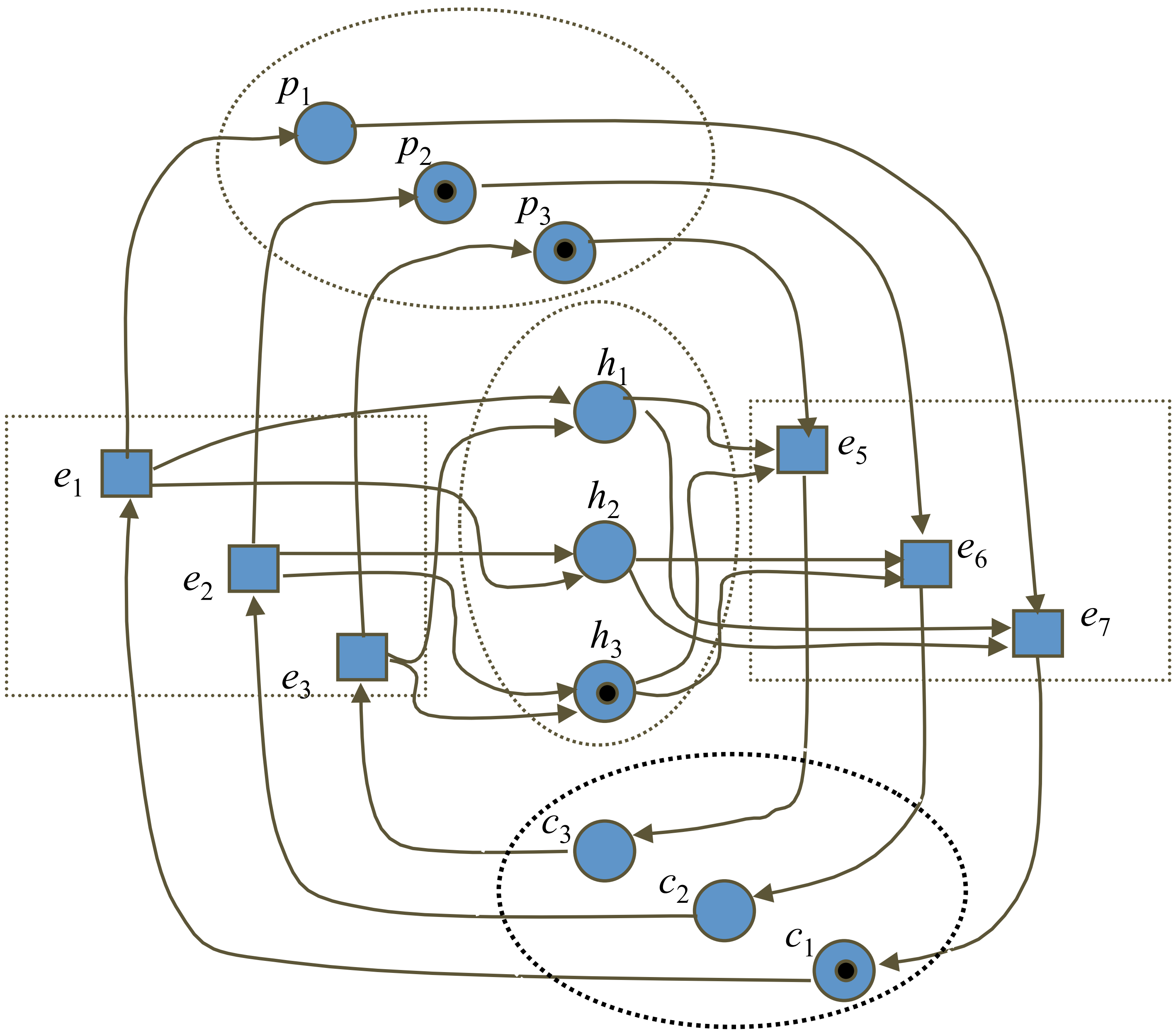
Cinco filósofos sentam em uma mesa e podem alternadamente pensar ou comer. No entanto, para esta última tarefa devem usar os utensílios ao lado do prato (hashis) o que impede os seus vizinhos de fazer a mesma coisa. O problema original prevê cinco filósofos sendo que somente dois deles conseguem passar do estado pensao para comendo simultaneamente.



Dobramento em RdP

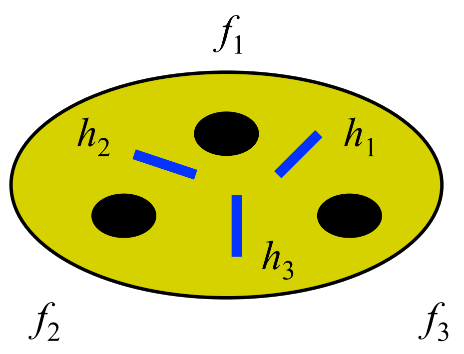
O exemplo dos filósofos

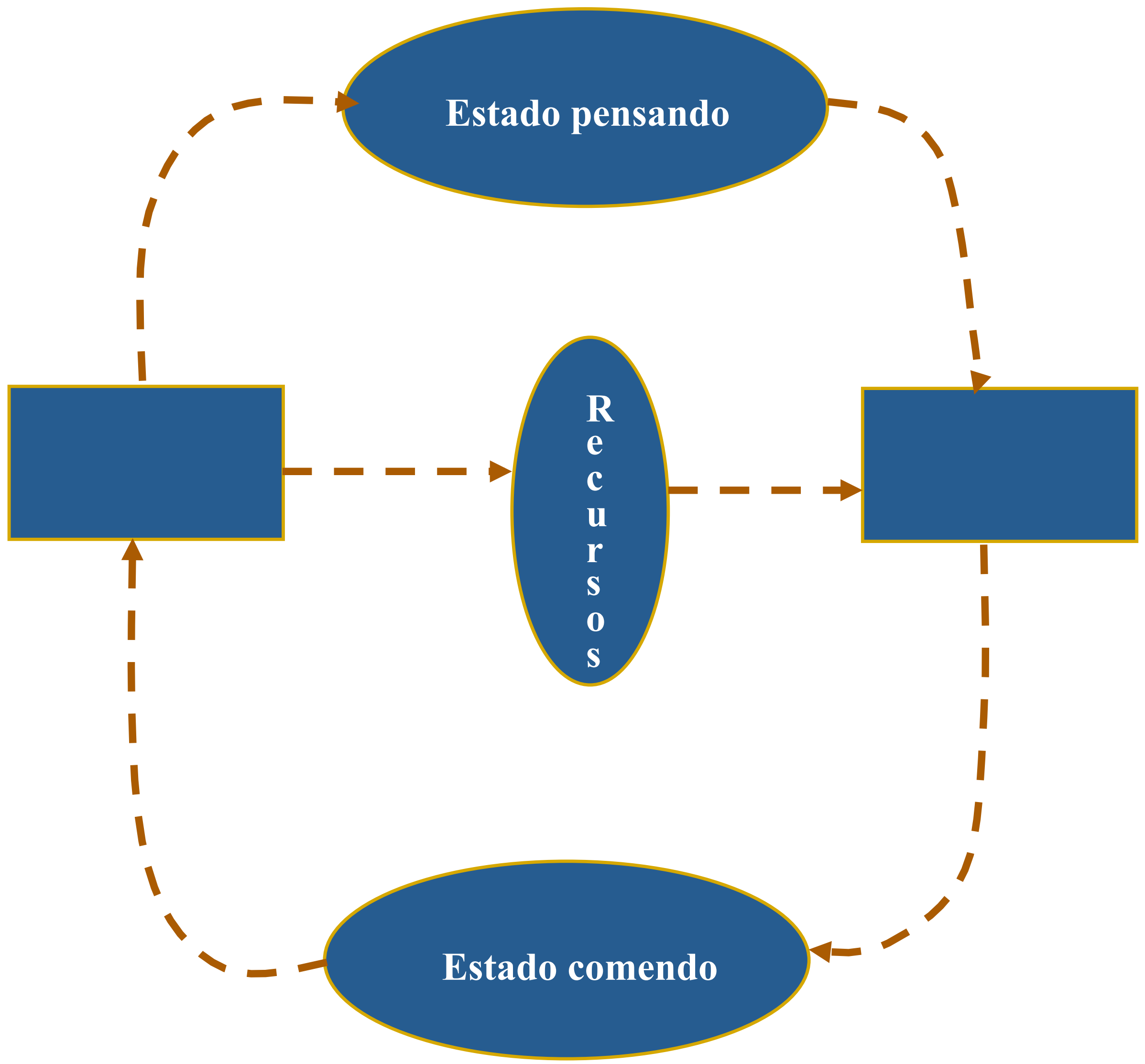




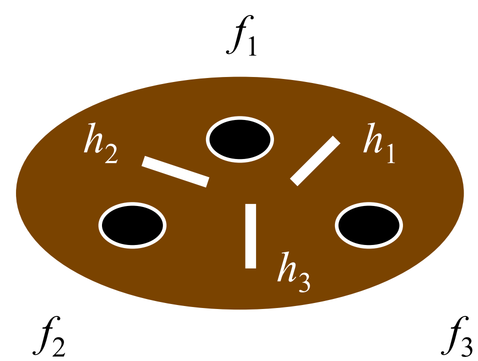
O dobramento

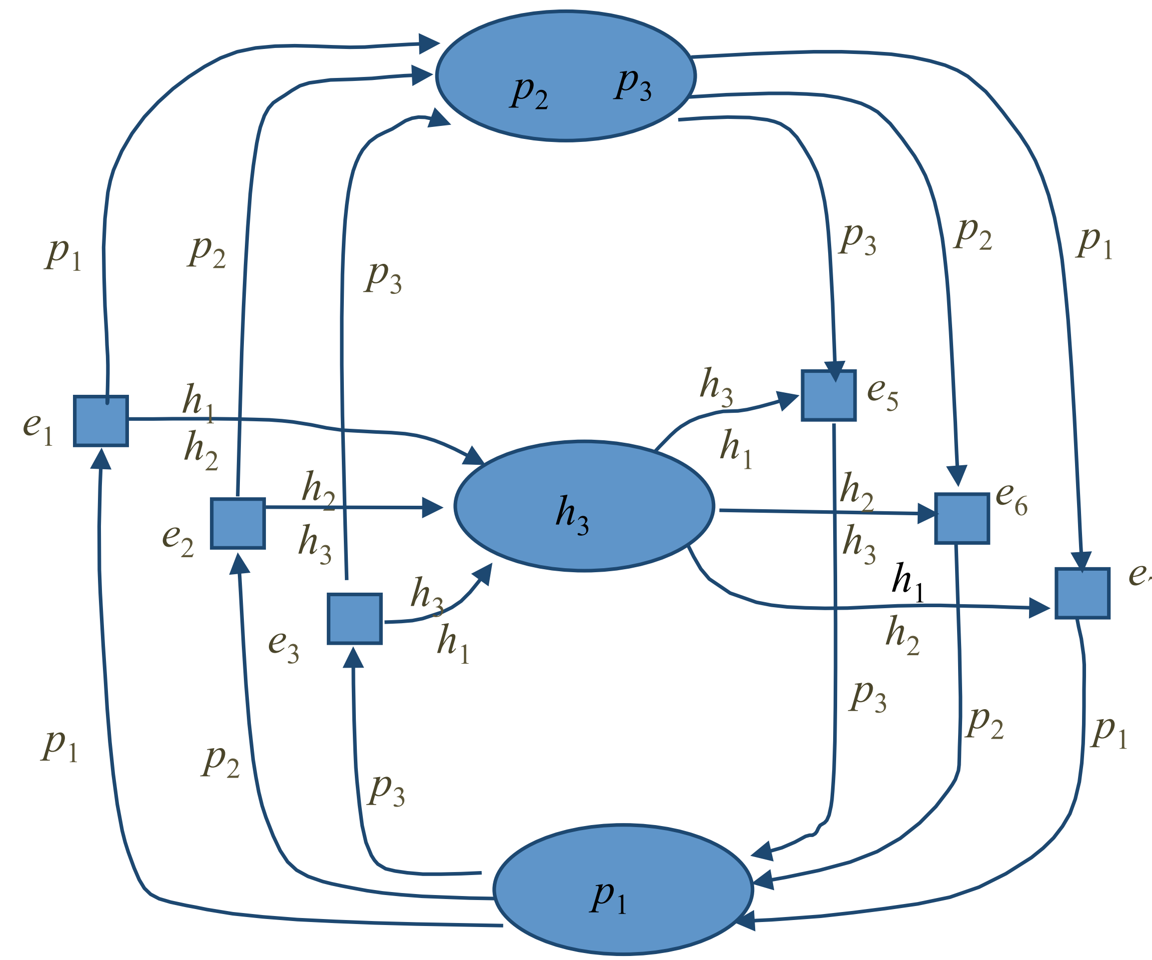
O exemplo dos filósofos





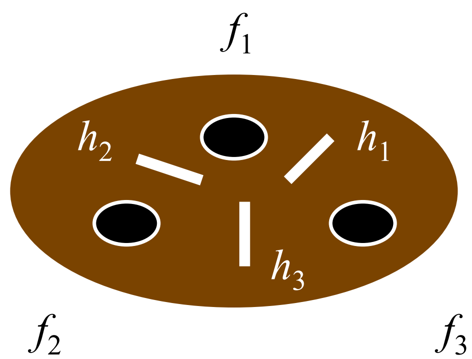
O exemplo dos filósofos





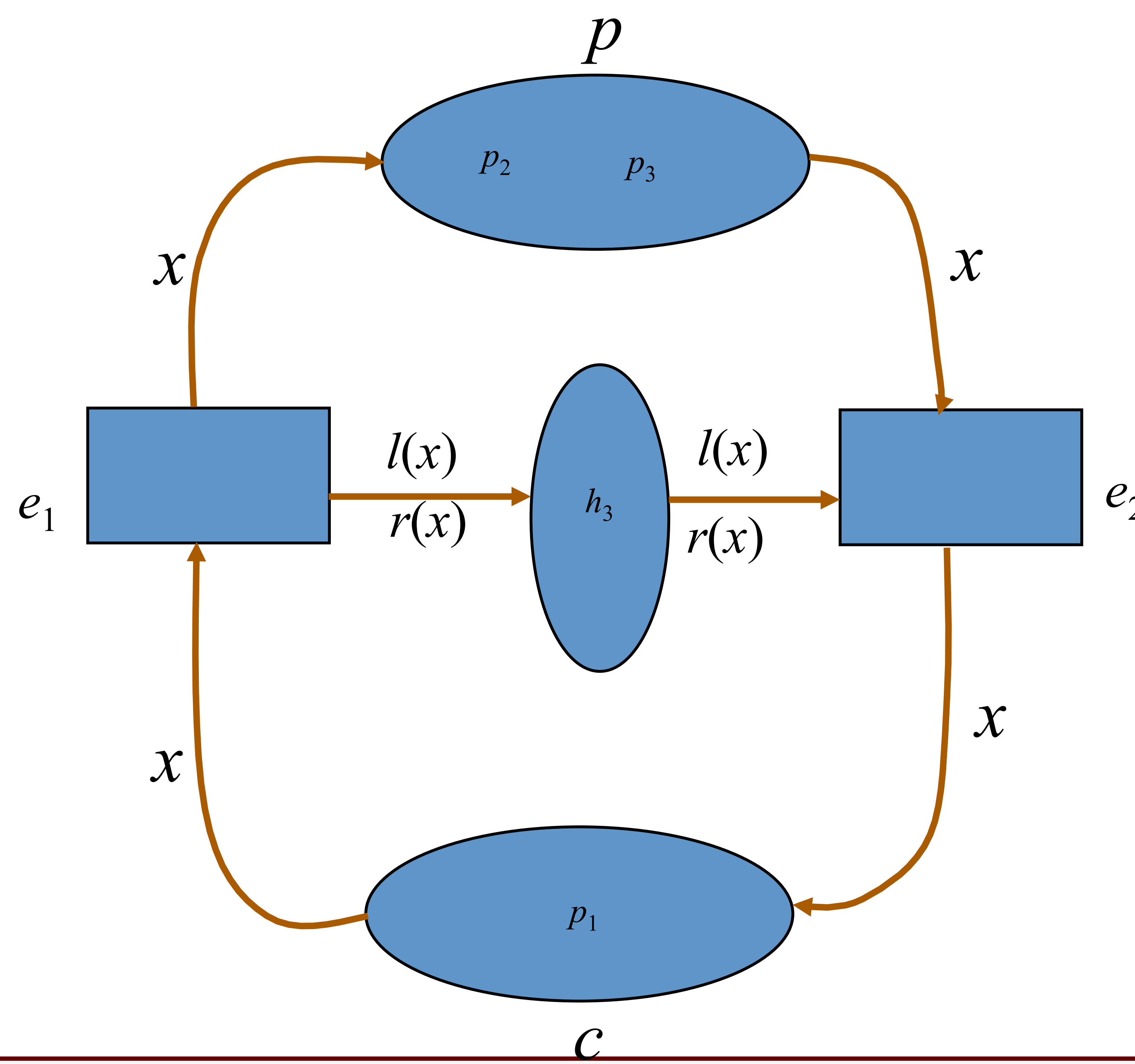
O colapso dos lugares leva à necessidade de se distinguir as marcas, tanto as que representam os filósofos quanto as que representam os recursos, isto é, os hashis.

O dobramento completo



$P=\{p_1, p_2, p_3\}$
$H=\{h_1, h_2, h_3\}$
$U=P \cup H$

$l : P \rightarrow H$
$p_i \rightarrow h_i$
$r : P \rightarrow H$
$p_1 \rightarrow h_2$
$p_2 \rightarrow h_3$
$p_3 \rightarrow h_1$



O problema da distinguibilidade

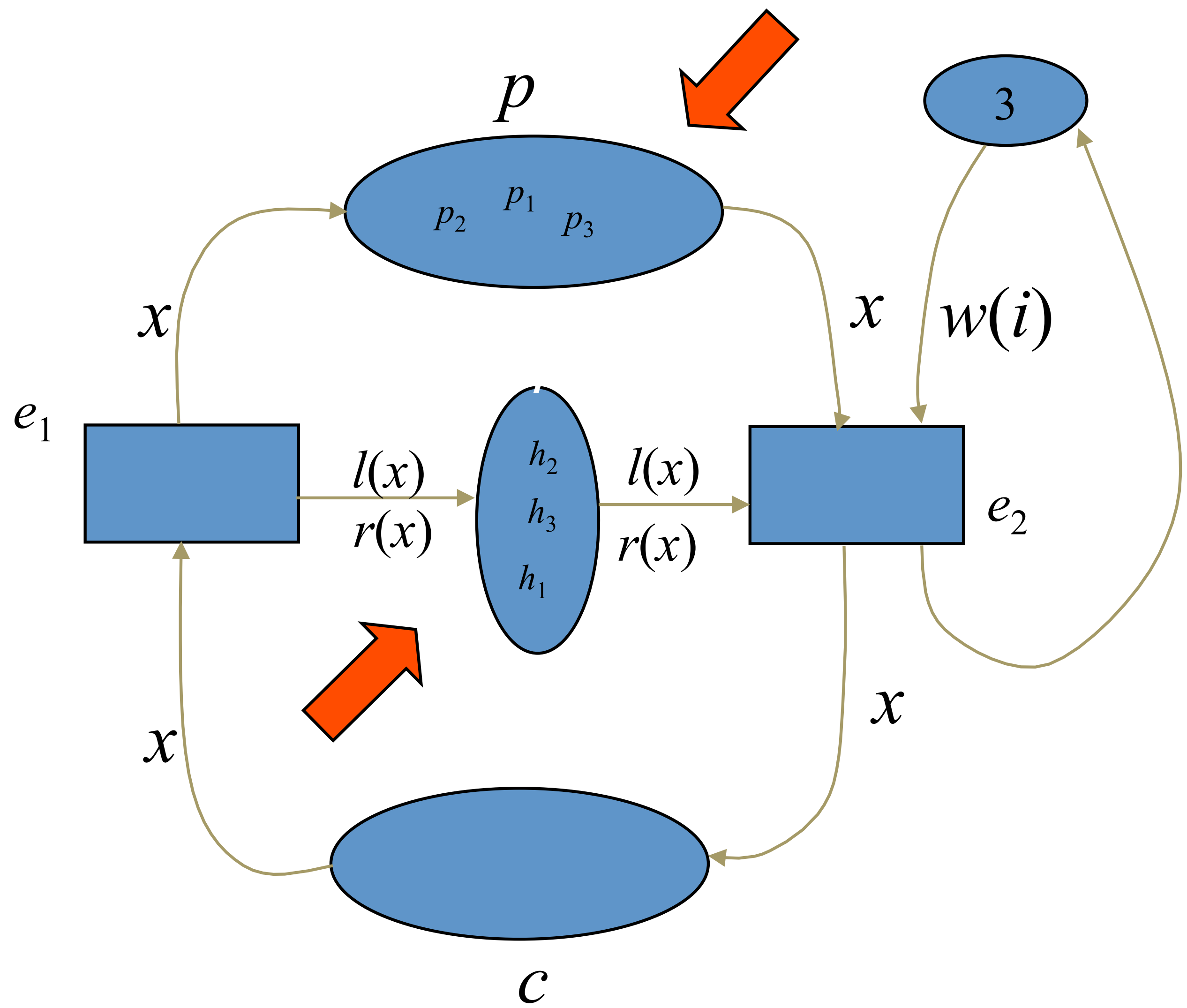
Como vimos anteriormente com o problema dos filósofos, a introdução da distinguibilidade das marcas (dobramento) não afeta as propriedades principais das redes.

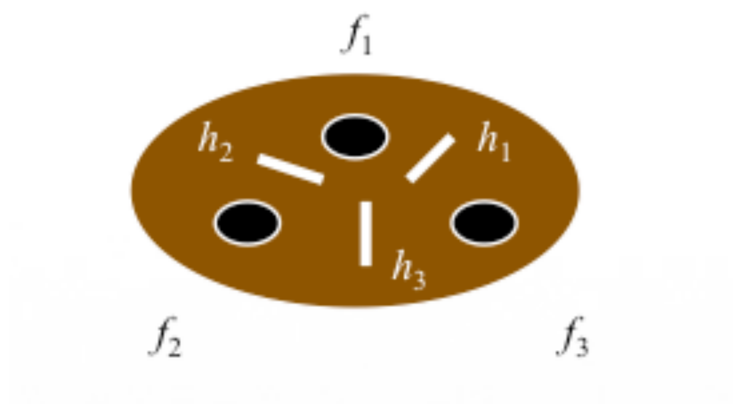
... isto é,

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

Não saímos do domínio das redes de Petri

Representação da marcação e multisets





A distinguibilidade das marcas nos leva à introdução de estruturas especiais no lugar das marcas, chamadas multisets.

Multisets

Seja o conjunto (base set) {a,b,c,d,e,f,g,h}.

Em teoria de conjuntos, $\{a, c, f\} \cup \{c, f, g\} = \{a, c, f, g\}$.

Já em multisets (bags),

$$\{a, c, f\} \cup \{c, f, g\} = \{a, c, c, f, f, g\}$$

também descrito como 1\`a + 2\`c + 2\`f + 1\`g

De fato, o mesmo conjunto pode ser escrito como,

1\`a + 0\`b + 2\`c + 0\`d + 2\`f + 1\`g + 0\`h

onde os coeficientes indicam o grau de multiplicidade de cada elemento do conjunto de base.

A forma simplificada para especificar este conjunto consiste em suprimir os elementos de coeficiente nulo.

Operações básicas

União : Sejam dois multisets representados pelos respectivos vetores de coeficientes sobre um conjunto de base (base set),

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

A união entre estes multisets é dada por,

$$m \cup n = (m_1 + n_1, \ldots m_i, n_j, \ldots, m_s + n_s)$$

Multiplicação por um escalar : Dado um multiset sobre um conjunto de base S, representado pelo seu vetor de coeficientes,

m= $(m_1, \ldots, m_i, \ldots, m_s)$.

Está definida a multiplicação de m por um escalar *p*, resultando no seguinte multiset,

$$p * \mathrm{m} = (p\, m_1, \ldots, p\, m_i, \ldots, p\, m_s)$$

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

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

Está definida a comparação entre estes dois multisets

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

Cardinalidade de um multiset : Um multiset

m= $(m_1, m_2, \ldots, m_i, \ldots, m_s)$,

tem cardinalidade $|m|= \Sigma\, m_i$

Subtração : Sejam dois multisets representados pelos respectivos vetores de coeficientes sobre um conjunto de base (base set),

m= $(m_1, m_2, \ldots, m_i, \ldots, m_s)$,
n= $(n_1, n_2, \ldots, n_i, \ldots, n_s)$

A subtração entre estes dois multisets, m - n, existe se m ≥ n e é dada por,

m - n = $(m_1 - n_1, \ldots, m_i - n_i, \ldots. m_s - n_s)$

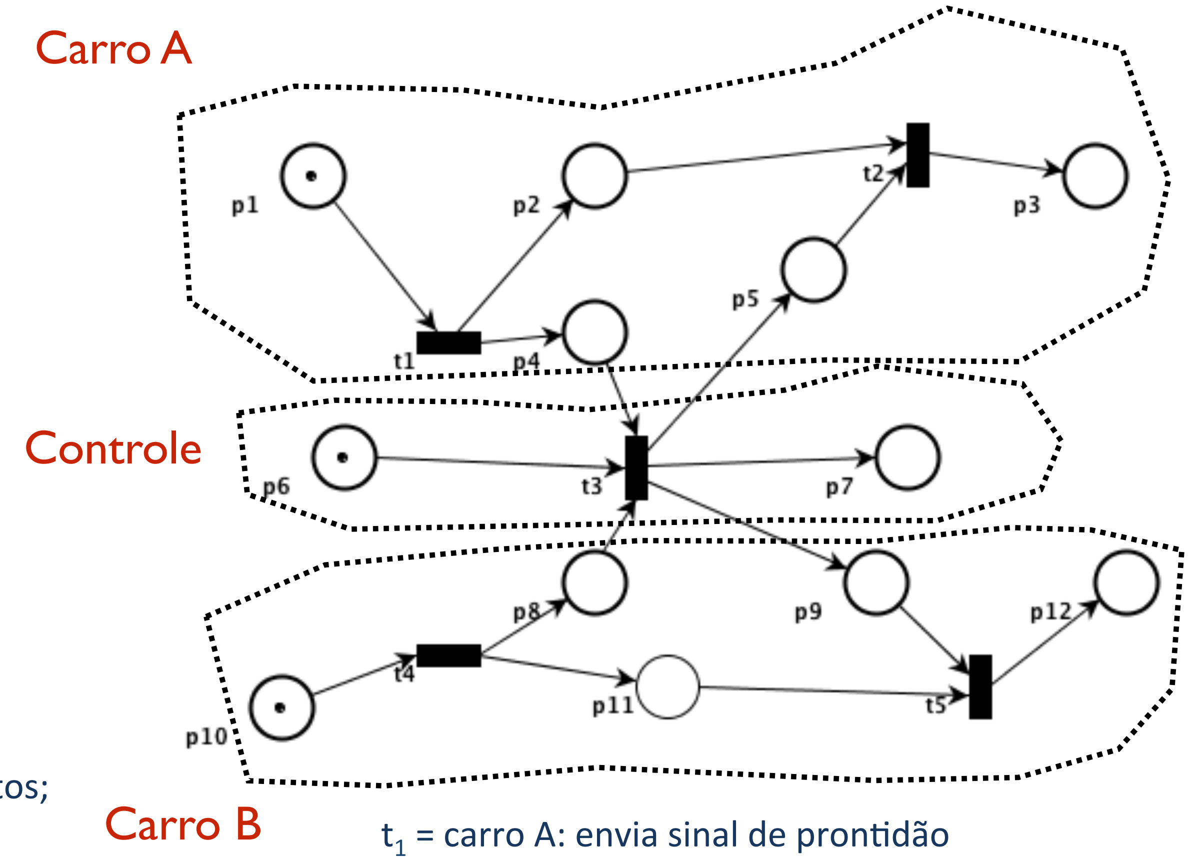
Se a forma de reconhecer simetria for sempre "visual", como alegam alguns autores, valeria a pena optar por uma representação em redes de alto nível?

Tipos de problemas

Redes clássicas

Redes de alto nível

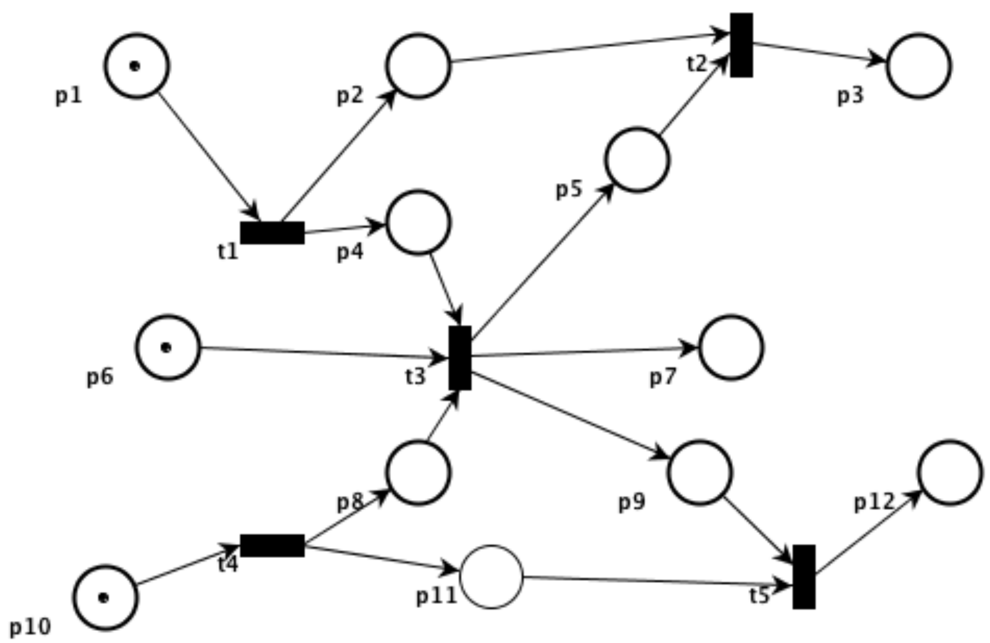
Voltando ao exemplo



p_1= carro A: preparando-se para começar;
p_2= carro A: esperando o sinal de largada;
p_3= carro A: correndo;
p_4= sinal de prontidão do carro A enviado;
p_5= sinal de largada para o carro A enviado;
p_6= operador: esperando sinal de prontidão dos pilotos;
p_7= operador: sinal de largada enviado;
p_8= sinal de prontidão do carro B enviado;
p_9= sinal de largada para o carro B enviado;
p_{10}= carro B: preparando-se para começar;
p_{11}= carro B: esperando o sinal de largada;
p_{12}= carro B: correndo;

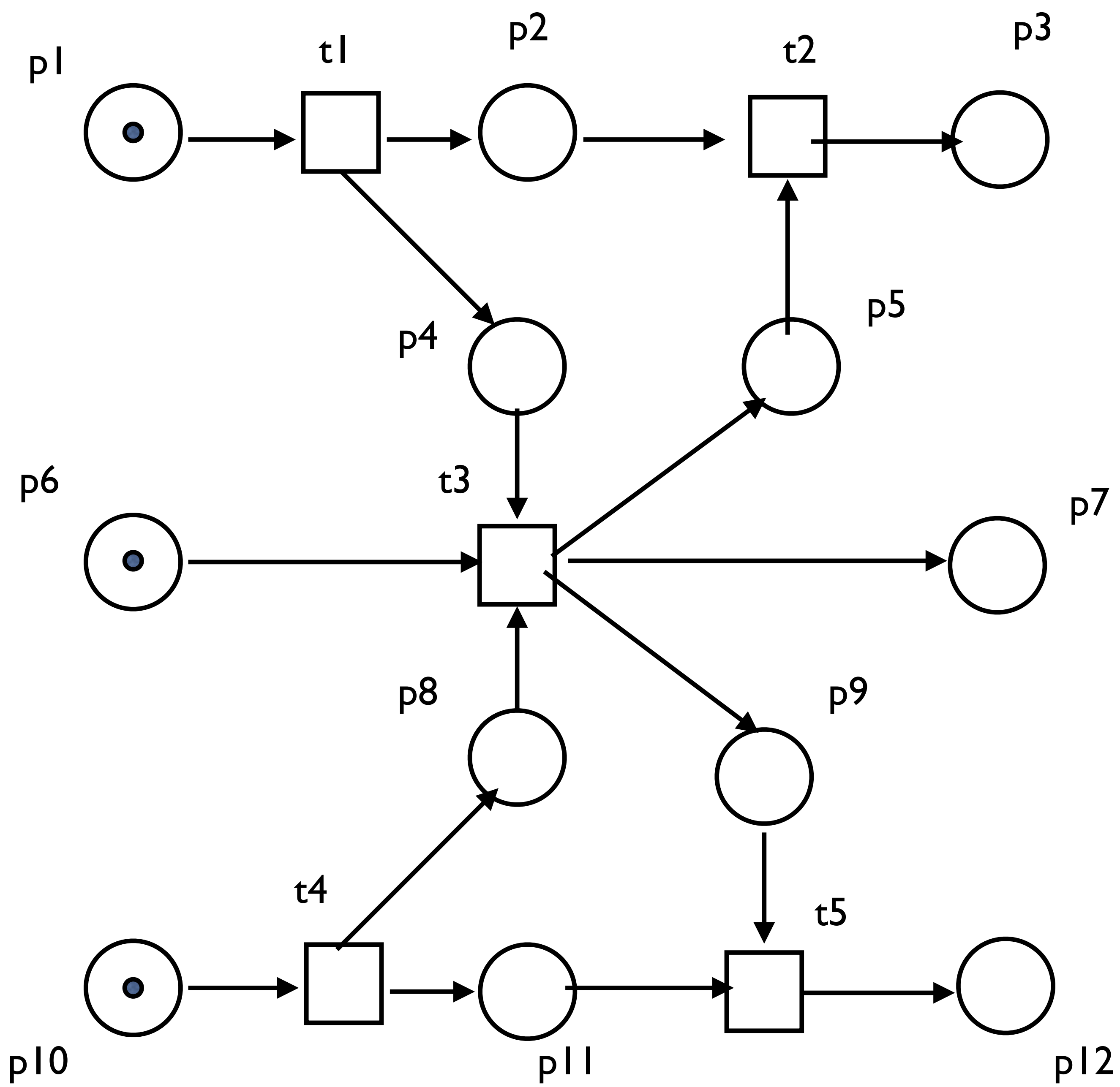
t_1 = carro A: envia sinal de prontidão
t_2 = carro A: acelera
t_3 = operador: manda sinal de largada
t_4 = carro B: envia sinal de prontidão
t_5 = carro B: acelera

Voltando ao exemplo

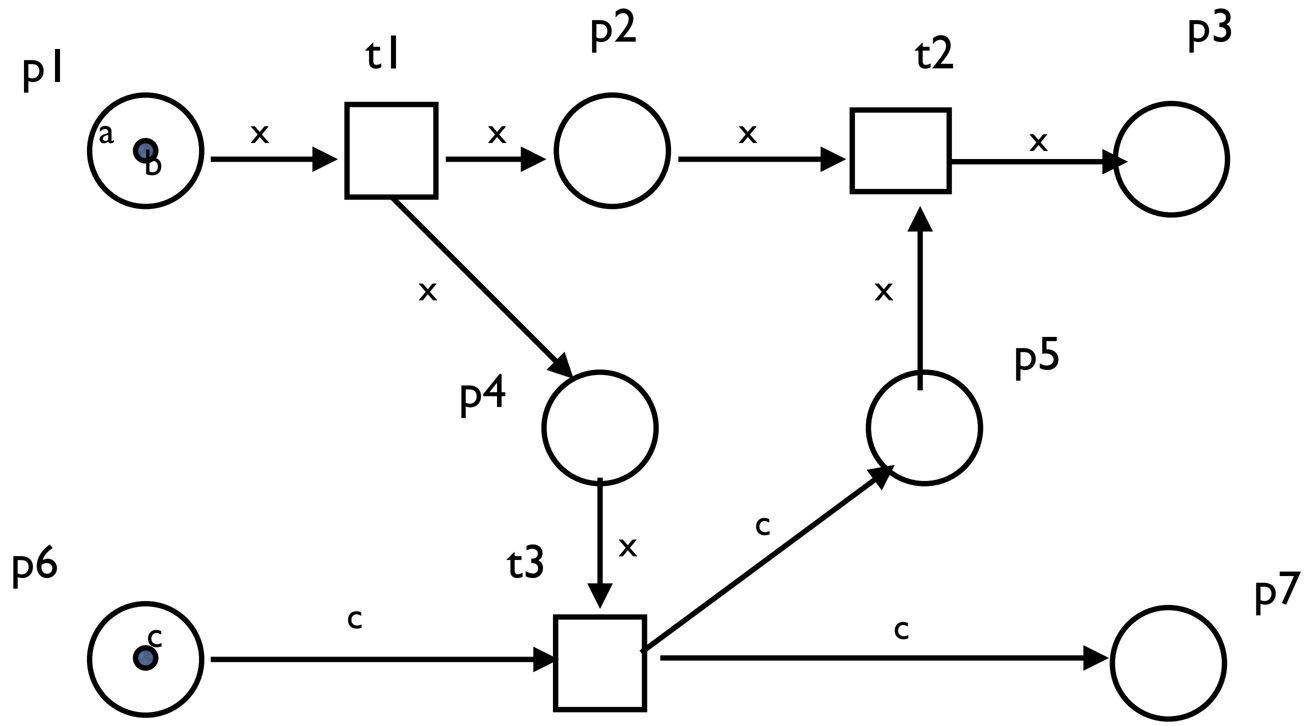


p_1= carro A: preparando-se para começar;
p_2= carro A: esperando o sinal de largada;
p_3= carro A: correndo;
p_4= sinal de prontidão do carro A enviado;
p_5= sinal de largada para o carro A enviado;
p_6= operador: esperando sinal de prontidão dos pilotos;
p_7= operador: sinal de largada enviado;
p_8= sinal de prontidão do carro B enviado;
p_9= sinal de largada para o carro B enviado;
p_{10}= carro B: preparando-se para começar;
p_{11}= carro B: esperando o sinal de largada;
p_{12}= carro B: correndo;

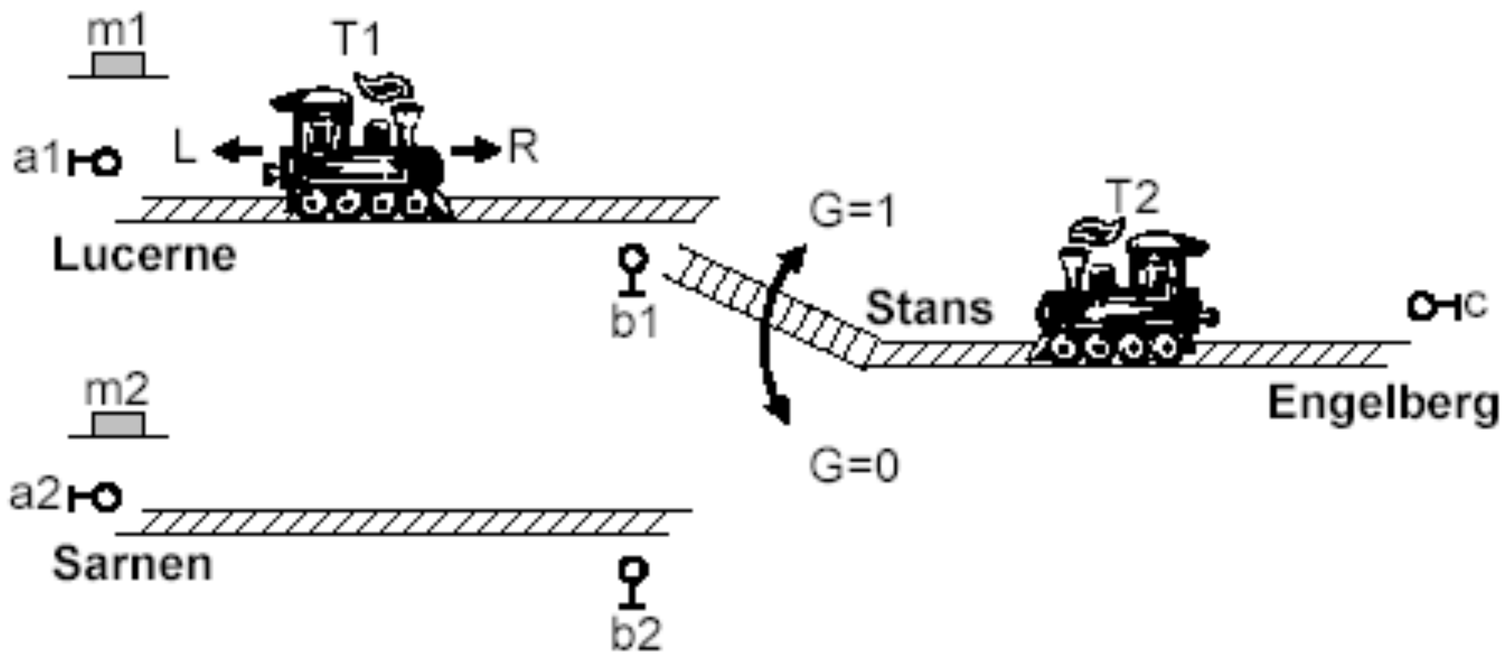
t_1 = carro A: envia sinal de prontidão
t_2 = carro A: acelera
t_3 = operador: manda sinal de largada
t_4 = carro B: envia sinal de prontidão
t_5 = carro B: acelera


p1
t1
p2
t2
p3
p4
p5
p6
t3
p7
p8
p9
t4
t5
p10
p11
p12

Cars = {a,b}
Cont = c



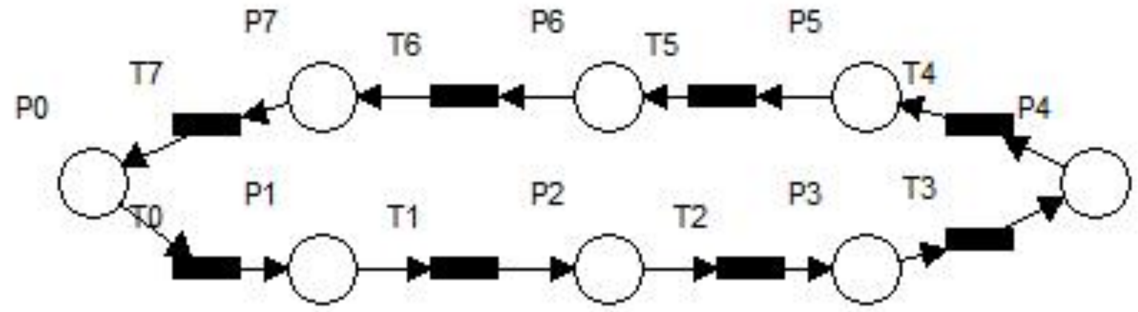
Exemplo: manobrando linhas de trem



Movimento do trem T1

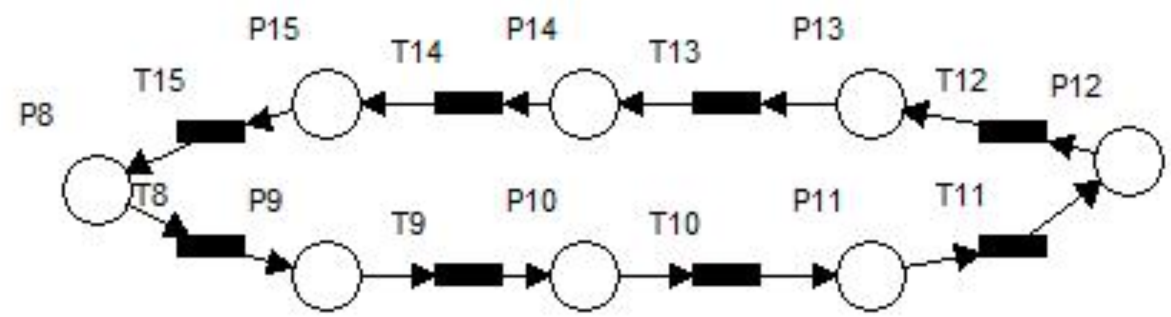
P1 – trem PT1 no ponto a1 (Lucerne);
P2 – trem T1 indo de Lucerne para Stans;
P3 – trem T1 chega em stans (detectado pelo sensor b1)
P4 – trem T1 no trecho unificado Stans Engelberg
P5 – trem T1 chega em Engelberg;
P6 – trem T1 indo de Engelberg para Stans (trecho unificado);
P7 – trem T1 chega no gate 1 (não há detecção por b1);
P8 – trem T1 indo de Stans para Lucerne;

Movement of Train T1

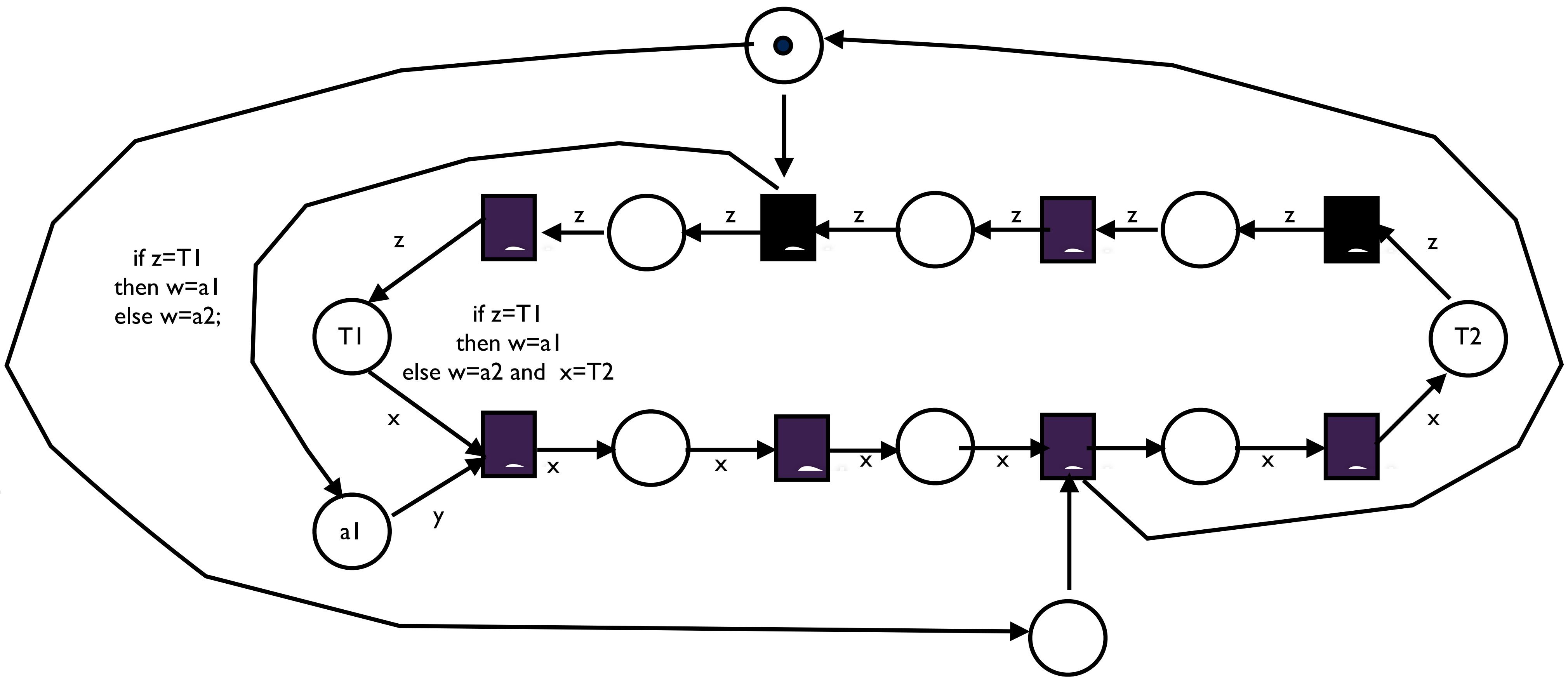


Movimento do trem T2

P9 – trem T2 no ponto C (Engelberg);
P10 – trem T2 indo de Engelberg para Stans;
P11 – trem T2 chega em Stans (não detectado pelo sensor b2)
P12 – trem T2 indo de Stans para Sarnen;
P13 – trem T2 chega em Sarnen;
P14 – trem T2 indo de Sarnen para Stans ;
P15 – trem T2 chega no gate 1 (Stans) (detectado pelo sensor b2);
P16 – trem T2 indo de Stans para Engelberg;



Mouvement of Train T2

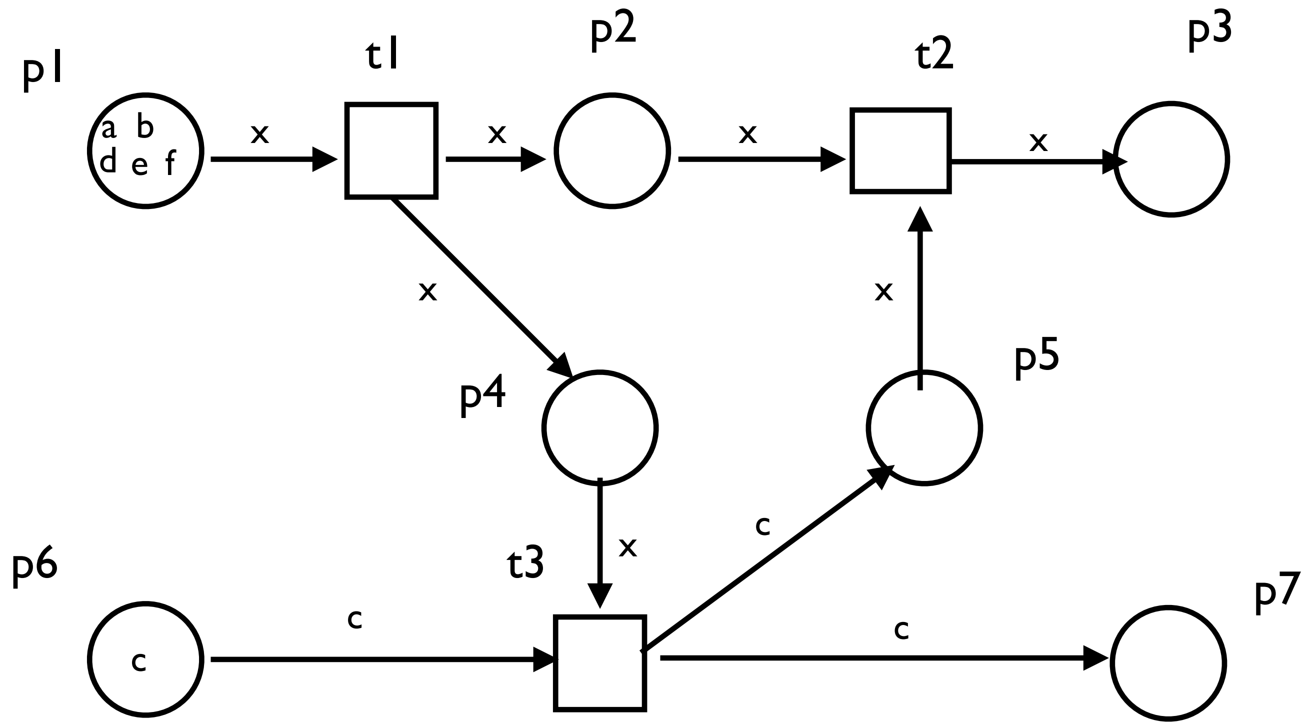


Train = {T1,T2}
op_sign = {a1, a2}
gate = {0, 1}
x, z in Train
w, y in op_sign

z
z

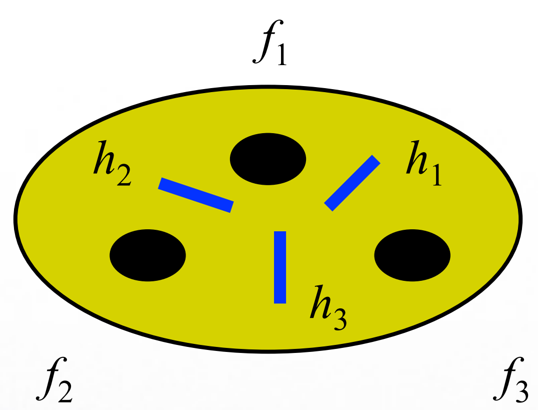
Valeu a pena o dobramento?

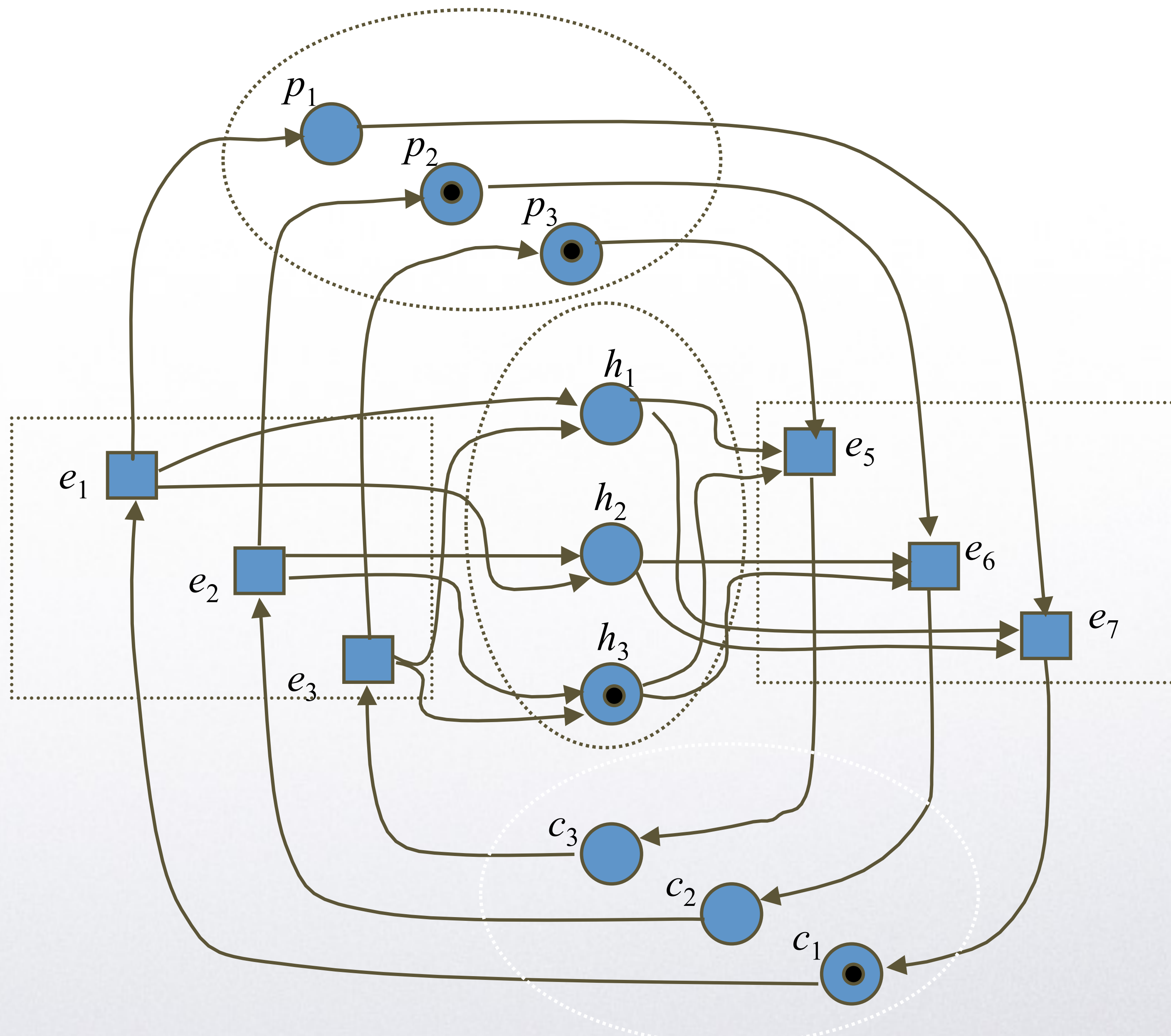
Cars = {a,b, d, e, f}
Cont = c



Dobramento em RdP

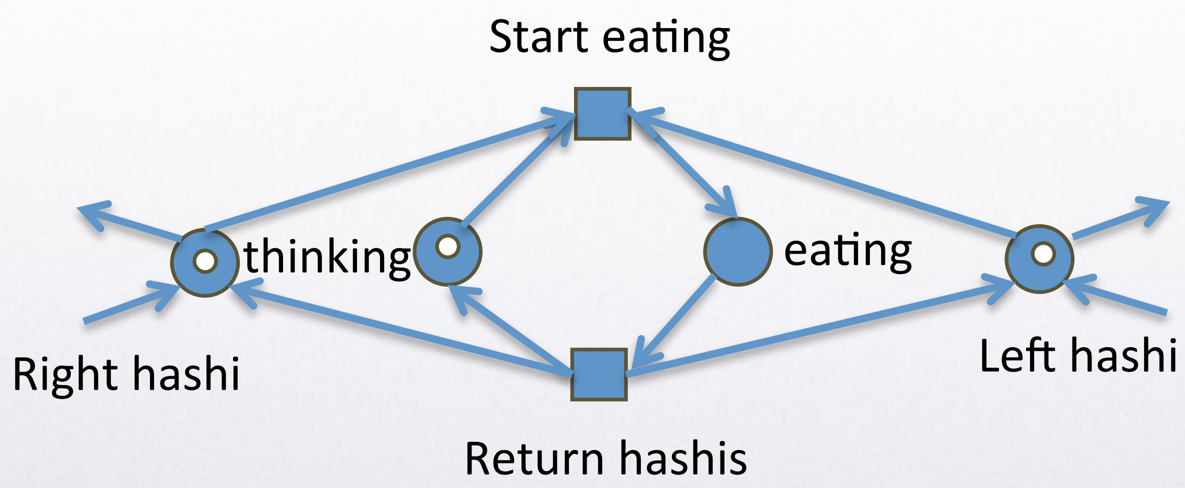
O exemplo dos filósofos





Modelagem e simetria

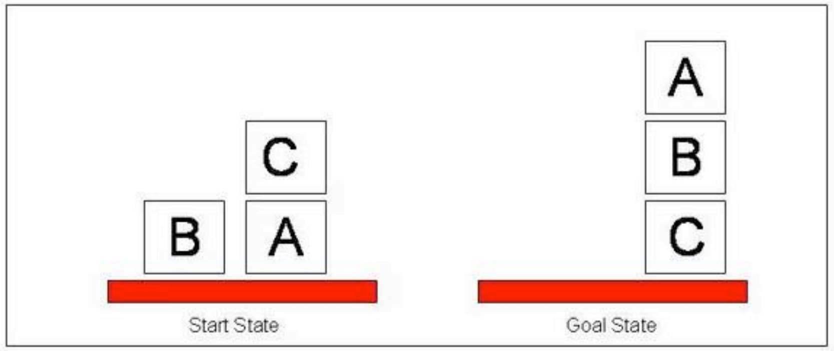
Certamente, uma forma de olhar o problema é procurar, logo de início, o relacionamento entre TODOS os seus elementos constituintes, como feito no slide anterior. Mas é possível também olhar cada um dos elementos composicionais, especialmente aqueles que apresentam propriedades reptetitivas. No exemplo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



Você conseguiria sintetizar a rede de alto nível (ou colorida) diretamente do enunciado do problema?

Tente fazer isso, mesmo já tendo visto a solução.

Voltando ao mundo dos blocos



The "Sussman anomaly" blocks-world planning problem.

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.

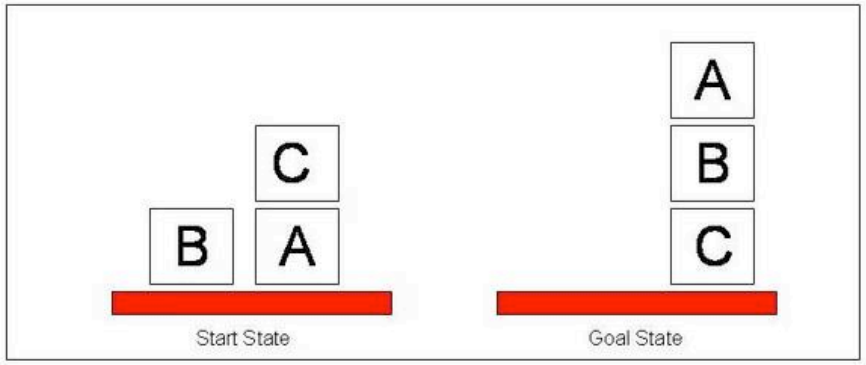
Com estas regras pretende-se fazer um plano de ações para que o robô transforme o estado inicial mostrado na figura no estado final.

Descrevendo os estados

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

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

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



The "Sussman anomaly" blocks-world planning problem.

Descrevendo as ações

1. **Pickup(x)**
 (robô pega um bloco x da mesa)
Pré-condições: ONTABLE(x), CLEAR(x), HANDEMPTY
Pós-condição: HOLDING(x);
2. **Putdown(x)**
(robô deposita bloco x na mesa)
Pré-condição: HOLDING(x)
Pós-condição: ONTABLE(x), CLEAR(x), HANDEMPTY
3. **Stack(x, y)**
(robô empilha bloco x sobre o bloco y)
Pré-condição: HOLDING(x), CLEAR(y)
Pós-condição: ON(x,y), HANDEMPTY, CLEAR(x)
4. **Unstack(x, y)**
(robô tira bloco x de cima do bloco y)
Pré-condição: ON(x,y), CLEAR(x), HANDEMPTY
Pós-condição: CLEAR(y), HOLDING(x)

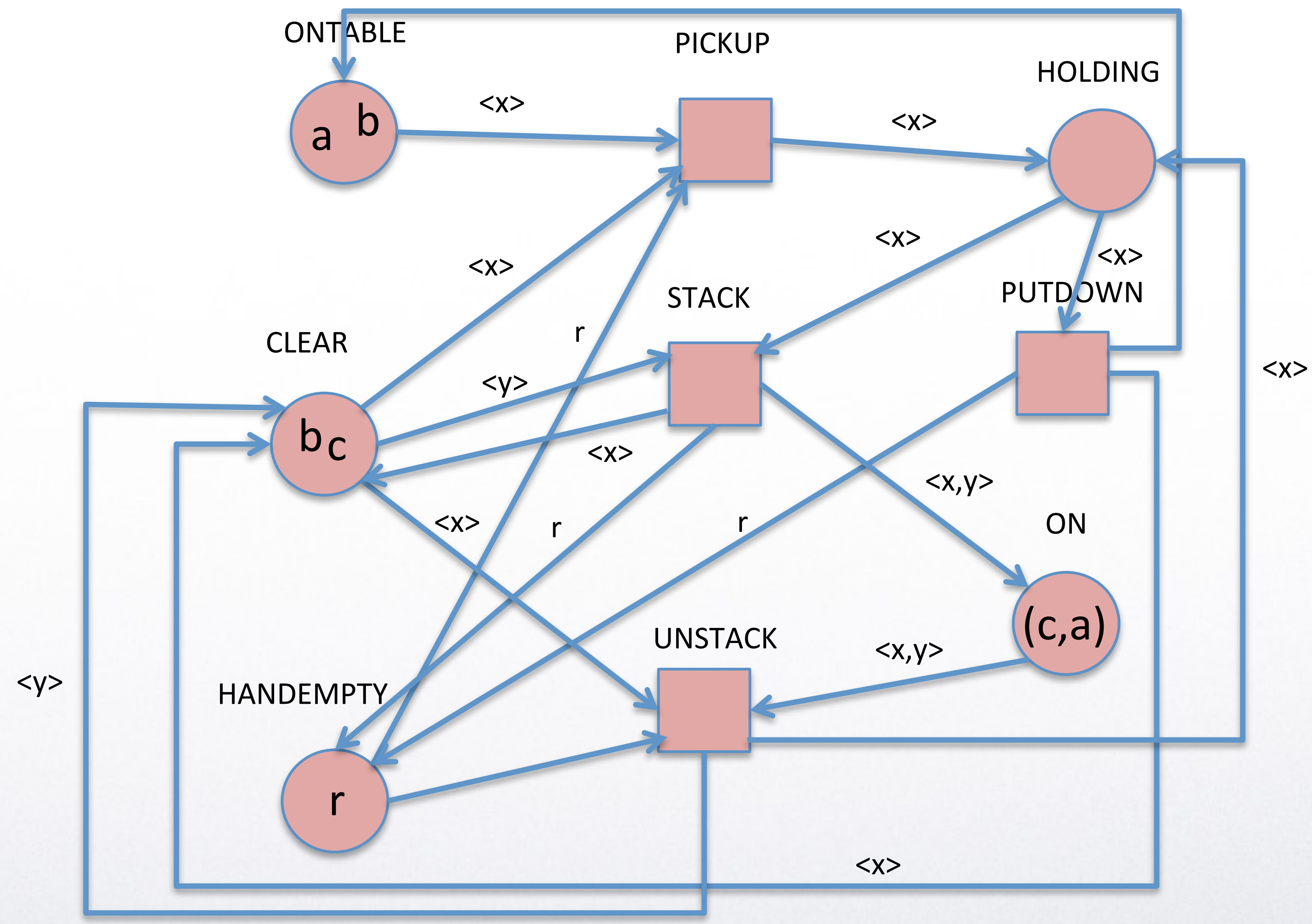
O problema (de planejamento)

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.

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

x,y: Blk



As alternativas

- Redes Coloridas (baseadas em conjuntos e na linguagem ML)
- Redes orientadas a objetos

- Redes estendidas

At a time when it is increasingly understood that programs must withstand rigorous analysis, particularly for systems where safety is critical, a rigorous language presentation is even important for negotiators and contractors; for a robust program written in an insecure language is like
a house built upon sand.

Robin Milner

As redes coloridas demarcam uma fase onde se buscava não apenas enfrentar a "explosão combinatória", do espaço de estados mas garantir o formalismo da modelagem.

Portanto as redes coloridas (e de alto nível) devem ser vistas como um convite a aumentar o nível de abstração dos processos de modelasse em Engenharia.

Voltando ao exercício de modelagem da lista…

Redes Coloridas (desfazendo os mitos)

As vantagens das redes coloridas seriam:

1. Explorar as simetrias com o objetivo de reduzir o tamanho da rede.

Falha: o grafo é reduzido mas a informação para para as inscrições a complexidade da representação não se altera significativamente

2. Ter uma representação abstrata para a rede clássica.

Falha: verdade em princípio, mas ter que reduzir a representação para rede clássica cada vez que deseja fazer análise de propriedades pode não ser exatamente uma vantagem.

3. Aumentar o poder de expressão da representação baseada em redes.

Falha: simplesmente não é verdade.

Histórico das redes CPN

As redes coloridas surgiram nos anos 80 conjugando a representação gráfica das redes de Petri com o Standard ML, que representa tipos e cores.

No final dos anos 80 e princípio dos anos 90 surgiu o ambiente Dsign CPN proposto pelo mesmo grupo de Ahus, Dinamarca (Kurt Jensen).

A idéia é simplesmente ter um formalismo mais abstrato

Definição informal das CPNs

Kurt Jensen

Coloured Petri Nets (CP-nets or CPNs) is a graphical language for constructing models of concurrent systems and analyzing their properties. CP-nets is a discrete-event modeling language combining Petri Nets and the functional programming language CPN ML which is based on Standard ML.

Aplicações

Aplicações Típicas

- Protocolo de Comunicação
- Redes de Dados
- Algorítmos Distribuídos
- Sistemas Embarcados

Novas Aplicações

- Sistemas de workflow
- Sitemas de manufatura
- Sistemas multi-agente
- Processos de negócio
- Análise de Requisitos

Uso prático das redes CPN

Como no caso das redes clássicas o uso prático das redes CPN está associado a **Simulação** do modelo, e portanto ao estudo de cenários específicos e ao processo de evolução das marcas.

Temos portanto o mesmo problema de desenvolver métodos alternativos de análise baseados nas propriedades da rede e fugir do problema da atingibilidade.

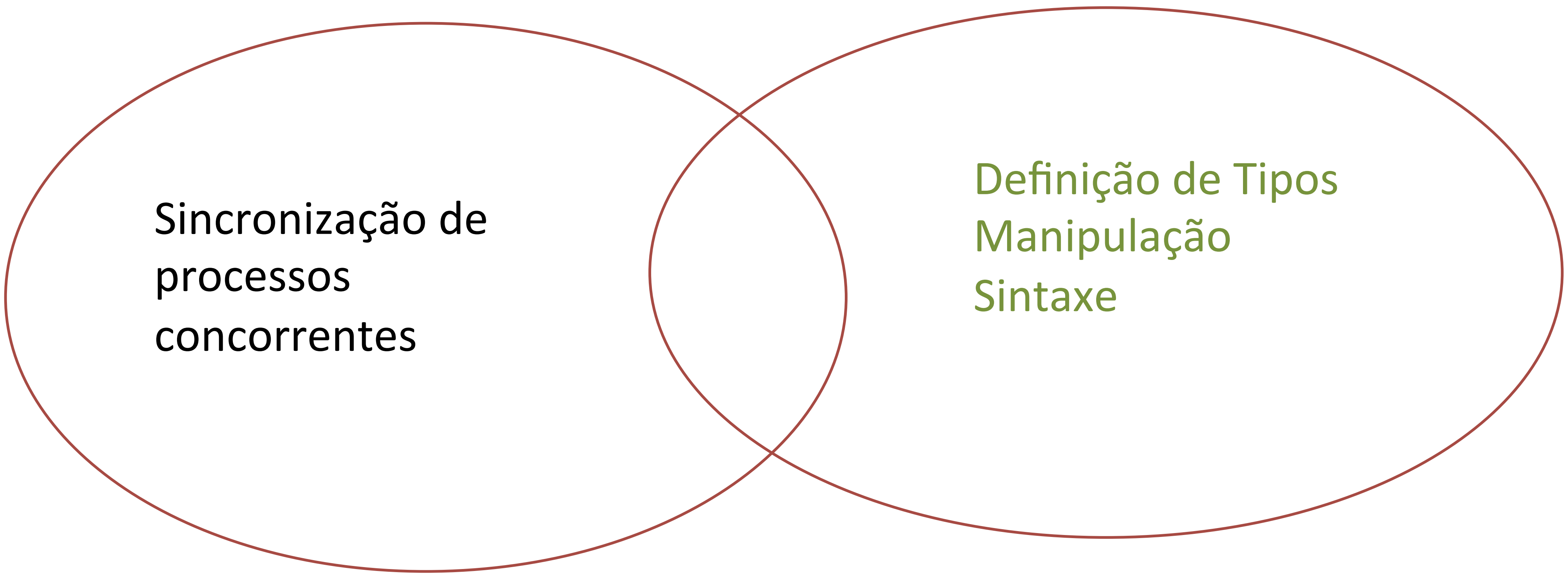
Novos métodos → Verificação e Model checking

Redes Coloridas

- As marcas são divididas em conjuntos e separadas por tipo
- A área de declaração do sistema contém a identidade de cada variável assim como em declarações em ML
- Os arcos possuem inscrições e filtros que selecionam o tipo de marca que pode fluir por este arco.
- O comportamento dinâmico é dado pelo conjunto : grafo, inscrições e declaração.

Redes de Petri Convencionais

Linguagens Formais



Kurt Jensen, An Introduction to the Practical Use of Coloured Petri Nets, Lect. Notes in Comp. Science 1492, 1998.

Processos especiais

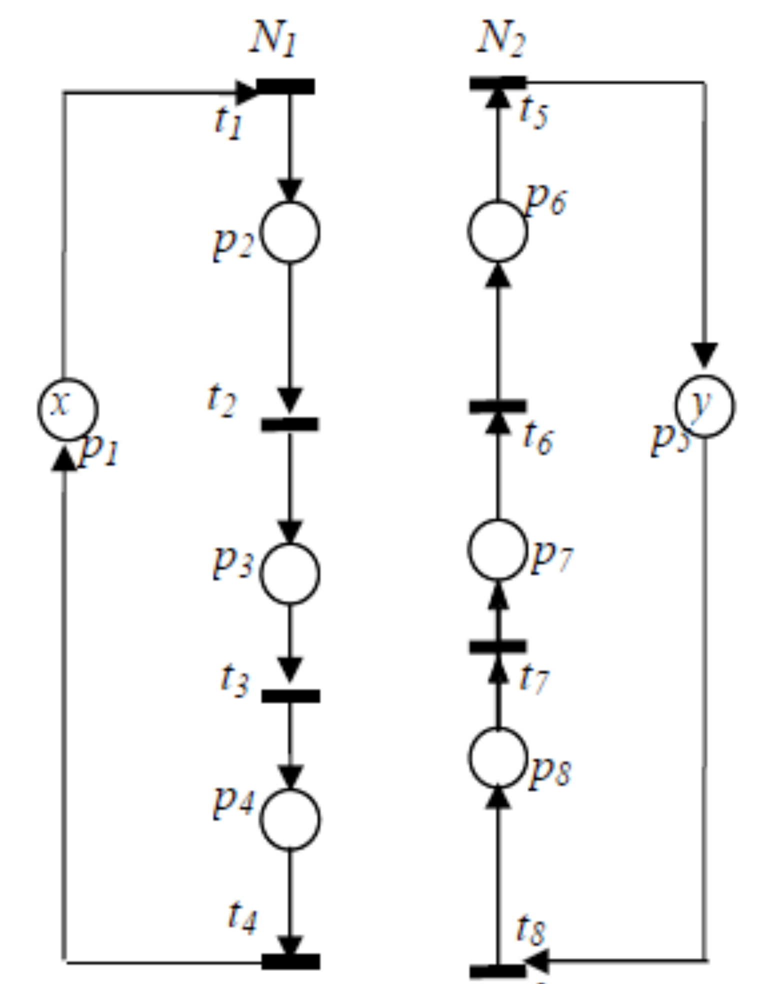


Fig. 1(b) Two S²P.

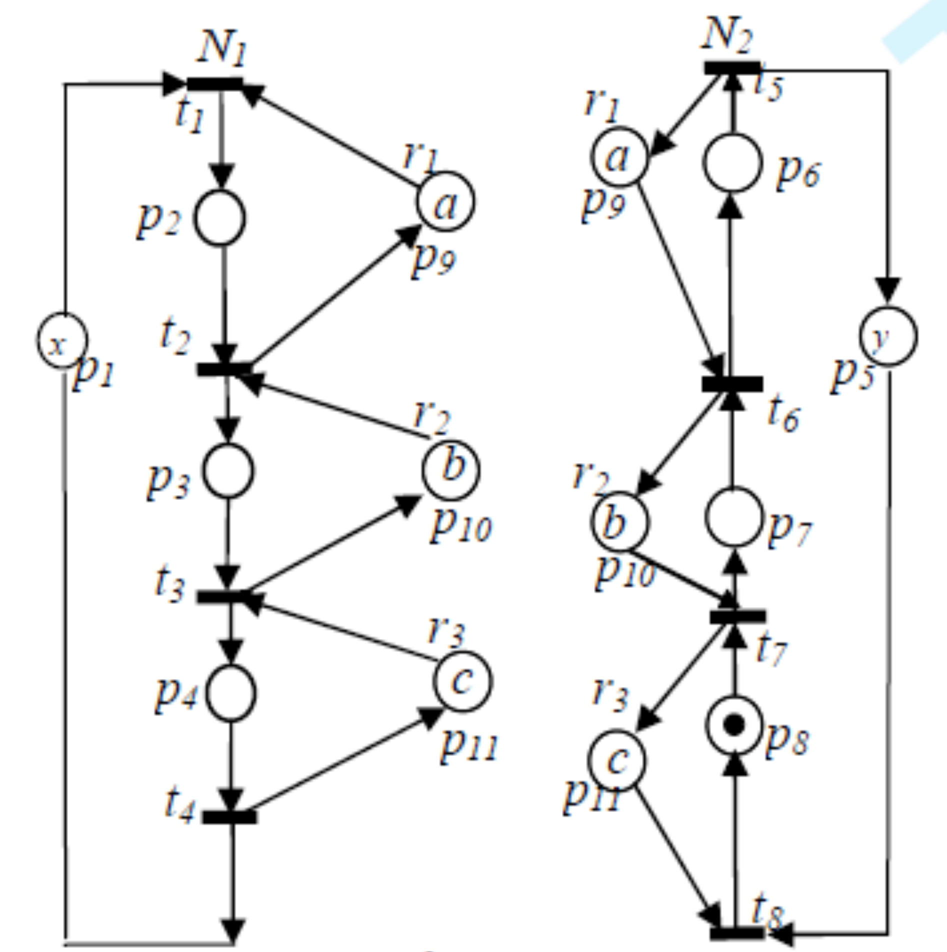


Fig. 1(c) S²PR of N_1 and N_2

O problema do acoplamento com recursos

O sistema S2PR pode ainda ser acoplado de modo que os processos sequenciais interferem um no outro podendo causar atrasos e até deadlocks devido à falta de recursos no momento devido.

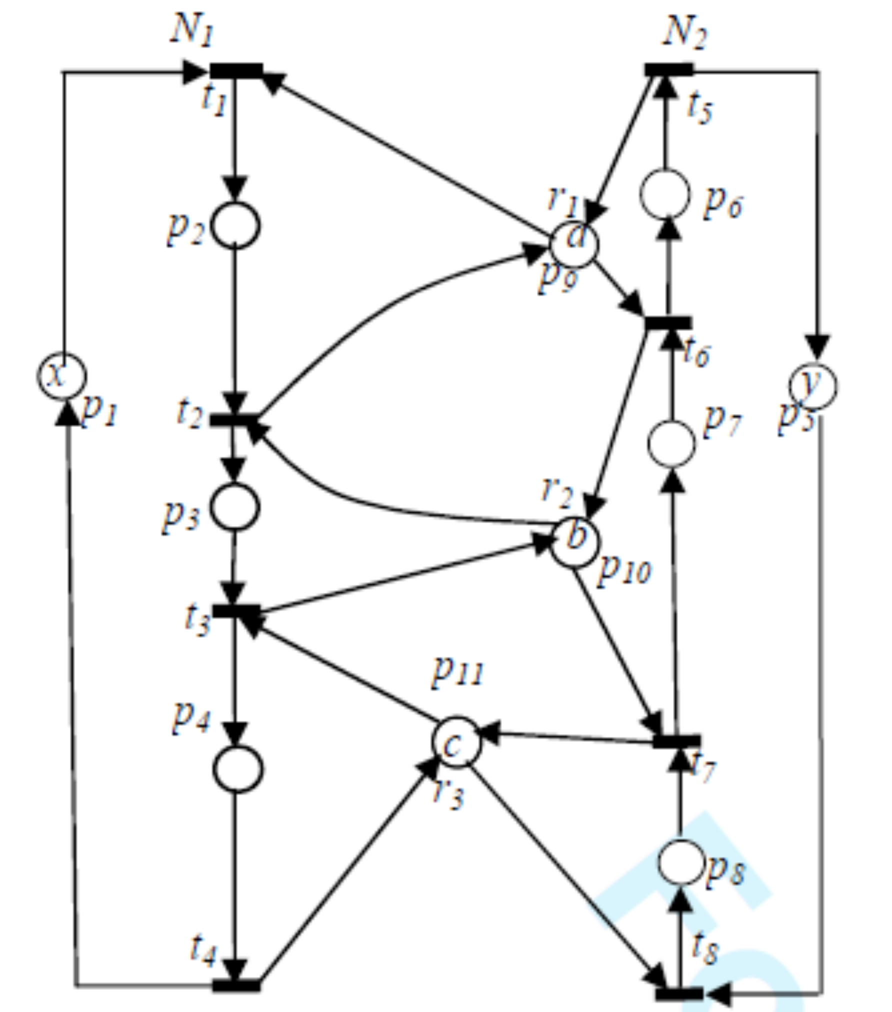


Fig. 1(a) S^3PR & 3^{rd}-order system ; $a=b=c=1$.

Li, Z.W. and Zhou, M.C., “Deadlock resolution in automated manufacturing systems: A novel Petri net approach,” Springer, London, 2009

Introdução informal: O problema da alocação de recursos

Na indústria automotiva moderna é comum se ter vários processos ou linhas de montagem e nestes um ou mais tipos de automóvel sendo montados em pipeline. Isso traz um problema, que é ter o tipo certo de insumo ou recurso no tempo correto, para a matriz correta.

Um problema similar e igualmente importante é ter processos homogêneos mas compartilhando o mesmo centro de recursos.

Fim