Redes de Petri Temporizadas

Prof. Eduardo Tavares
Prof. Paulo Maciel
Centro de Informática (UFPE)

Disciplina: Modelos de Sistemas Comunicantes

Tempo

Várias definições

Tempo em sistemas computacionais (Interpretações)

- Tempo Lógico: definido a partir de relações de precedência entre eventos. Estabelece ordens causais entre conjunto de eventos.
- Tempo Físico: tempo métrico que expressa quantitativamente a distância entre eventos. Estabelece também as ordens totais entre eventos
- Tempo Contínuo: segue a natureza uniforme e contínua do tempo físico e é isomorfo a $\Re$
- Tempo Discreto: simplificação do tempo contínuo e isomorfo a $\aleph$

Farine e et al. Sistemas de Tempo Real. 2000

Tempo

- Tempo Global: Referência temporal única para os componentes do sistema
- Tempo Local: Cada componente do sistema possui sua própria referência temporal

Qual a motivação da adoção de tempo nos modelos de sistemas comunicantes?

Farine e et al. Sistemas de Tempo Real. 2000

Avaliação de Desempenho

Measuring

- ➢Medição
- ➢*Benchmark*
- ➢Prototipação

Modelagem

- ➢Modelos de Simulação
- ➢Modelos Analíticos

Avaliação de Desempenho

Modelagem

- Modelos Analíticos
  - Determinísticos
    - Avaliação de pior (melhor) caso
  - Probablísticos
    - Valores médios prováveis
- Simulação
  - Análise Exaustiva

Implementação real

- Medidas obtidas do sistema real
- *Benchmark*
- Protótipos

Modelos Temporizados

Diversos modelos propostos. Alguns representativos (Probabilísticos e Deterministícos):

- Lógicas Temporais (Ex: Linear Time Temporal Logic)
- Autômatos temporizados
- Álgebra de processos temporizadas (ex: Timed CSP)
- Redes de Fila
- Cadeias de Markov
- Redes de Petri Temporizada (Ex: TPN)

Importância dos tempos físicos em sistemas críticos

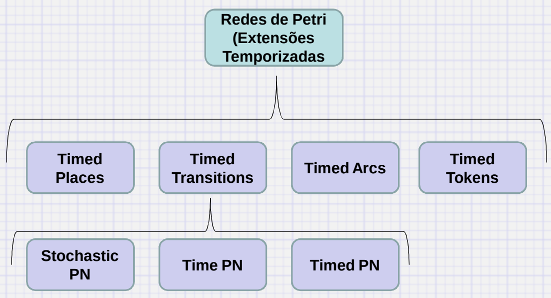
Foco será nos modelos determinísticos

Modelos Temporizados

Os modelos, que possibilitam a especificação do tempo físico, podem representar os tempos de formas distintas:

- Intervalo
- De forma determinística
- Forma probabilística (não é considerado nesta disciplina). Distribuição exponencial geralmente adotada.

Redes de Petri Temporizadas



Redes de Petri Temporizadas

Breve Histórico:
- Ranchandani, 1973 – Transition Timed Net
- Merling, 1976 – Transition Time Net
- Sifakis, 1977 – Place Timed Net

Extensões estocástica (*Delay é uma variável aleatória de distribuição exponencial)*
- Natkin, 1980
- Moloy, 1981
- Marsan et al., 1984

Lugares Temporizados

Tempo associados com lugares

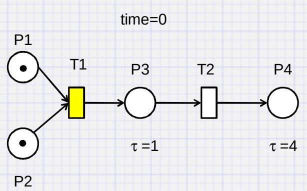
Tokens ficam disponíveis nos lugares de saída após a passagem de um tempo especificado

Classificação dos tokens: disponíveis e indisponíveis

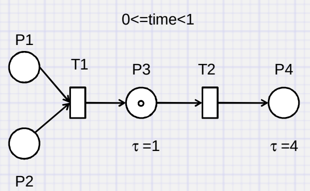
Tokens disponíveis habilitam transições

Conceito de *Holding Durations*

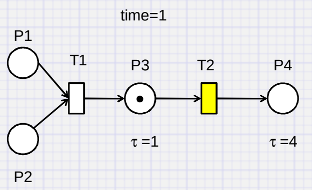
Lugares Temporizados



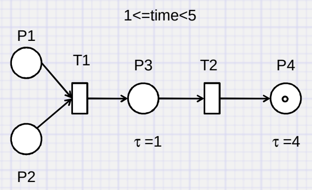
Lugares Temporizados



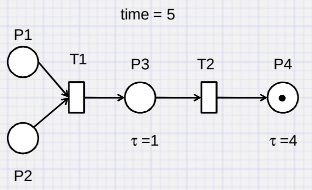
Lugares Temporizados



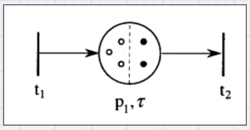
Lugares Temporizados



Lugares Temporizados



Lugares Temporizados

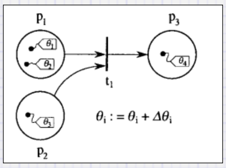


Tokens temporizados

Tempo associado com os tokens

Token guarda *timestamp* (indica quando uma transição pode ser disparada)

Timestamp pode ser incrementado ao disparo de uma transição

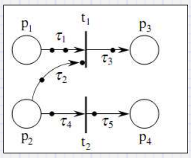


Arcos temporizados

Tempo associado com os arcos

Travelling delay é associado aos arcos

Tokens ficam indisponíveis até alcançar a transição



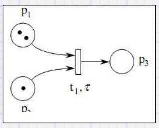
Transições Temporizadas

Extensão mais comum

Tempo associado com transições. Reresentação natural.

- Início da atividade com a habilitação da transição
- Término da atividade com o disparo da transição

O *delay* pode ser um valor constante ou intervalo



Transições Temporizadas (Conceitos Básicos)

Políticas de disparo

- Duração (Disparo em três fases)
  - *Tokens* (marcas) são consumidas dos lugares de entrada
  - Há uma duração
  - Tokens são gerados nos lugares de saída
- Disparo atômico
  - As marcas permanecem nos lugares de entrada pelo período igual ao *delay* associado à transíção
  - Após o delay, as marcas consumidas são imediatamente geradas nos lugares de saída

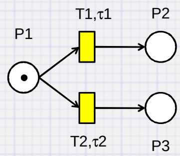
Transições Temporizadas (Conceitos Básicos)

Políticas de disparo

- ➢ Duração (Disparo em três fases)
  - ❑ O estado é uma informação mais complexa do que o modelo não temporizado
- ➢ Disparo atômico
  - ❑ O conjunto de marcações alcançáveis é um subconjunto das marcações do modelo sem temporização
  - ❑ Pode representar um modelo com duração

Transições Temporizadas (Conceitos Básicos)

Regras de Seleção



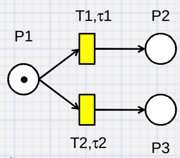
- ➢ Pré-seleção: (duração e delay)
  - ❑ Prioridade
  - ❑ Probabilidade
- ➢ Race(Corrida): (delay)
  - ❑ Transições com menor *delay* são disparadas

Transições Temporizadas (Conceitos Básicos)

Quando uma das transições conflitantes é desabilitada pelo disparo da outra, o que acontece com o timer daquela que ficou desabilitada?

Transições Temporizadas (Conceitos Básicos)

Como fica a memorização do tempo de habilitação anterior ?



<u>Mecanismos Básicos de Memória</u>

- ➢ Continue: O timer da transição mantém o valor atual

e quando a transição se tornar novamente habilitada o valor do timer iniciará naquele valor

- ➢ Restart: Quando a transição for novamente habilitada o *timer* será reiniciado

Transições Temporizadas (Conceitos Básicos)

O que acontece com o timer das transições habilitadas após o disparo de uma transição? (Para todas as transições, não somente as conflitantes)

Políticas de memória

- Resampling
  - Em todos os disparos de transições, os *timers* de todas as transições são descartadas (***restart***)
  - Nenhum histórico do passado é mantido
  - Na nova marcação, um novo valor para o timer é associado para cada transição habilitada

Transições Temporizadas (Conceitos Básicos)

Políticas de memória

- Enabling Memory
  - A cada disparo de uma transição, os *timers* das transições desabilitadas na nova marcação são descartados (***restart***)
  - O valor dos *timers* de todas transições que continuam habilitadas na nova marcação são mantidas (***continue)***

Transições Temporizadas (Conceitos Básicos)

Políticas de memória

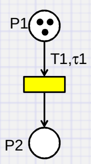
- Age Memory
  - Após cada disparo de uma transição, os *timers* mantém seus respectivos valores (***continue***), tanto para as transições habilitadas e desabilitadas na nova marcação

Transições Temporizadas (Conceitos Básicos)

Semântica de Temporização

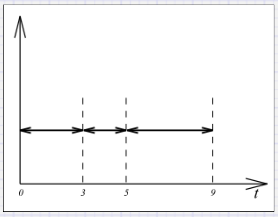
Qual procedimento deve-se realizar quando o grau de habilitação de uma transição é maior que 1?

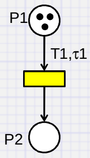
- Single-server firing semantics
- Infinite-server firing semantics
- Multiple-server firing semantics



Transições Temporizadas (Conceitos Básicos)

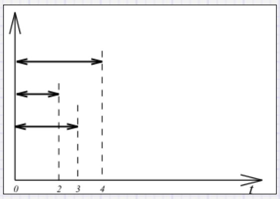
- Single-server firing semantics

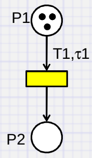




Transições Temporizadas (Conceitos Básicos)

- Infinite-server firing semantics

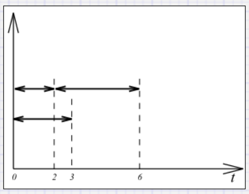


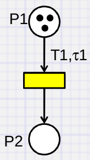


Transições Temporizadas (Conceitos Básicos)

- Multiple-server firing semantics

K = Grau máximo de paralelismo. Assuma K=2.



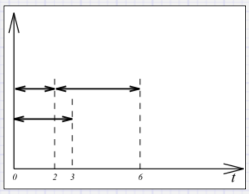


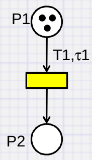
Se K=∞, então igual a infinite-server firing semantics

Transições Temporizadas (Conceitos Básicos)

- Multiple-server firing semantics

K = Grau máximo de paralelismo. Assuma K=2.





Se K=∞, então igual a infinite-server firing semantics

Leitura

L. Motus. Time Concepts in Real-Time Software. *Control Engineering Practice,* 1993.

F. D. J. Bowden. A Brief Survey and Synthesis of the Roles of Time in Petri Nets. *Mathematical and Computter Modelling,* 2000.

G. Balbo. Introduction to Stochastic Petri Nets. *Formal Methods on Performance Evaluation,* 2001.

Seção: Time in Petri Nets.

Time Petri Nets

Definição de Tavares09 e Barreto05 baseada em Merling76

Restrições temporais associado às transições (intervalo). Assume-se tempo discreto

Transições habilitadas – *enabled* (marcação) e disparáveis – *firable* (marcação e tempo)

Política *Enabling Memory*

Singler-server semantics e *Strong Firing Mode*

Time Petri Nets

Definition 3.6 (Petri net). *A Place/Transition net (Petri net) is a bipartite directed graph represented by a tuple* (P, T, F, W, m_0), *where* P *(set of places) and* T *(set of transitions) are non-empty disjoint sets of nodes* $(P \cap T = \emptyset)$. *The edges are represented by* F, *where* $F \subseteq A = (P \times T) \cup (T \times P)$. $W : A \to \mathbb{N}$ *represents the weight of the edges, such that*

$$W(f) = \begin{cases} x \in \mathbb{N}, & if\ (f \in F) \\ 0, & if\ (f \notin F) \end{cases}$$

A marking m_i *is a function* $(m_i : P \to \mathbb{N})$, *and* m_0 *is the initial marking.*

Definition 3.13 (Time Petri net). *A time Petri net is defined by a tuple* $(\mathcal{N}, I)$, *where* $\mathcal{N}$ *is the underlying Petri net, and* $I : T \to \mathbb{N} \times \mathbb{N}$ *represents the timing constraints, such that* $I(t) = (EFT(t), LFT(t))\ \forall t \in T$, $EFT(t) \leq LFT(t)$. $EFT(t)$ *is the Earliest Firing Time, and* $LFT(t)$ *is the Latest Firing Time.*

Definition 3.7 (Enabled Transitions). *A set of enabled transitions at marking* m_i *is denoted by:* $ET(m_i) = \{t \in T \mid m_i(p_j) \geq W(p_j, t)\}, \forall p_j \in P$.

Time Petri Nets

Vetor de clocks $c \in (\aleph \cup \{\#\})^{|T|}$

Dynamic Firing Interval: $I_D(t)$ = (DLB(t),DUB(t))

- DLB(t) = max(0,EFT(t)-c(t))
- DUB(t) = LFT(t) – c(t)

Atenção *Strong Firing Mode!*

Inicialmente, I(t)= $I_D(t)$

Time Petri Nets

Definition 3.14 (States). *Let $\mathcal{N}_\mathcal{T}$ be a time Petri net, $M \subseteq P \times \mathbb{N}$ be the set of reachable markings of $\mathcal{N}_\mathcal{T}$, and $C \subseteq (\mathbb{N} \cup \{\#\})^{|T|}$ be the set of clock vectors. The set of states S of $\mathcal{N}_\mathcal{T}$ is given by $S \subseteq (M \times C)$, that is, a state is defined by a marking, and the respective clock vector.*

Definition 3.15 (Firable Transitions). *Let $\mathcal{N}_\mathcal{T}$ be a time Petri net, the set of firable transitions at state $s \in S$ is defined by: $FT(s) = \{t_i \in ET(m) | \ DLB(t_i) \leq \min(DUB(t_k)), \forall t_k \in ET(m)\}$.*

$\boxed{FT \subseteq ET \subseteq T}$

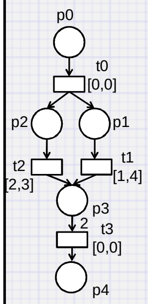
Definition 3.16 (Firing Domain). *The firing domain for a transition t at state s, is defined by the interval: $FD_s(t) = [DLB(t), \min(DUB(t_k))], \forall t_k \in ET(m)$.*

Time Petri Nets

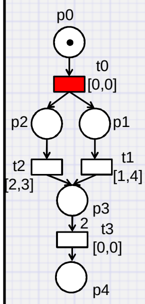
Definition 3.17 (Reachable States). *Let $\mathcal{N}_\mathcal{T}$ a time Petri net, and $s_i = (m_i, c_i)$ a reachable state. $s_j = \texttt{fire}(s_i, (t, \theta))$ denotes that firing a transition $t \in FT(s_i)$ at time $\theta \in FD_{s_i}(t)$ from the state s_i, the reached state $s_j = (m_j, c_j)$ is obtained from:*

- $\forall p \in P, \ m_j(p) = m_i(p) - W(p,t) + W(t,p)$, *as usual in Petri nets;*
- $\forall t_l \notin ET(m_j), c_j(t_l) = \#;$
- $\forall t_k \in ET(m_j), \ c_j(t_k) = \begin{cases} 0, & if(t_k = t) \\ 0, & if(t_k \in ET(m_j) - ET(m_i)) \\ c_i(t_k) + \theta, & else \end{cases}$

Time Petri Nets

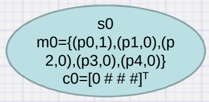


Time Petri Nets

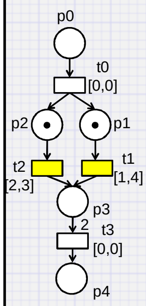


FT(s0)={t0}
c_{s0} (t0)=0

$I_D s_0$ (t0)=[0,0]
FDs_0(t0)=[0,0]



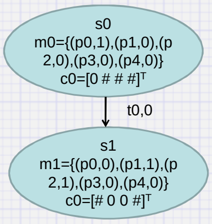
Time Petri Nets



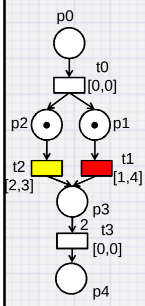
ET (m1)={t1,t2}
c_{s1} (t1)=0
c_{s1} (t2)=0

$I_D s_1$ (t1)=[1,4]
$I_D s_1$ (t2)=[2,3]

FDs_1(t1)=[1,3]
FDs_1(t2)=[2,3]



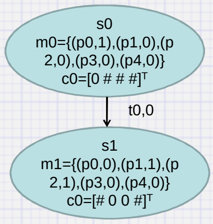
Time Petri Nets



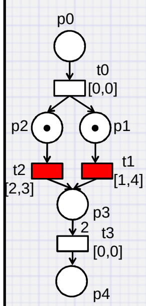
ET(m1)={t1,t2}
FT (s1)={t1}
c_{s1} (t1)=1
c_{s1} (t2)=1

$I_D s_1$ (t1)=[0,3]
$I_D s_1$ (t2)=[1,2]

FDs_1(t1)=[1,3]
FDs_1(t2)=[2,3]



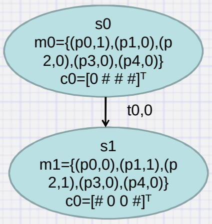
Time Petri Nets



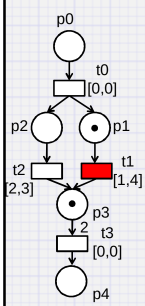
FT (s1)={t1,t2}
c_{s1} (t1)=2
c_{s1} (t2)=2

$I_D s_1$ (t1)=[0,2]
$I_D s_1$ (t2)=[0,1]

FDs_1(t1)=[1,3]
FDs_1(t2)=[2,3]

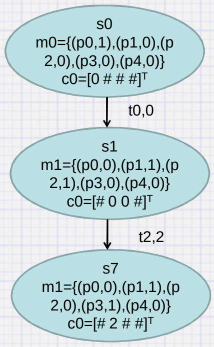


Time Petri Nets

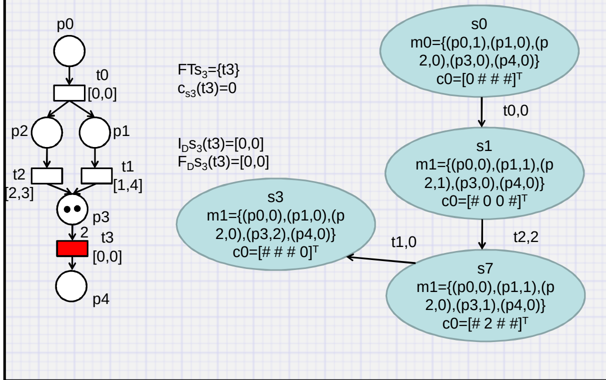


FTs_7={t1}
C_{s7}(t1)=2

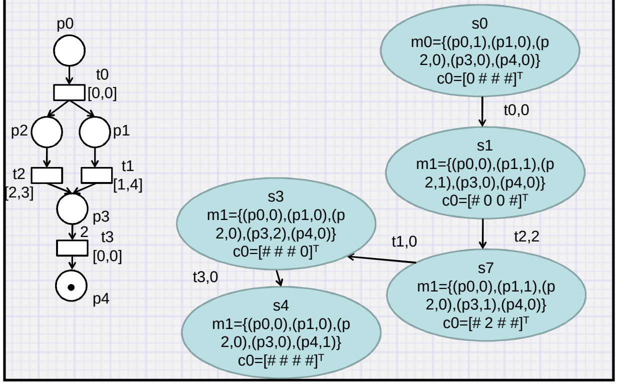
$I_D s_7$ (t1)=[0,2]
$F_D s_7$ (t1)=[0,2]

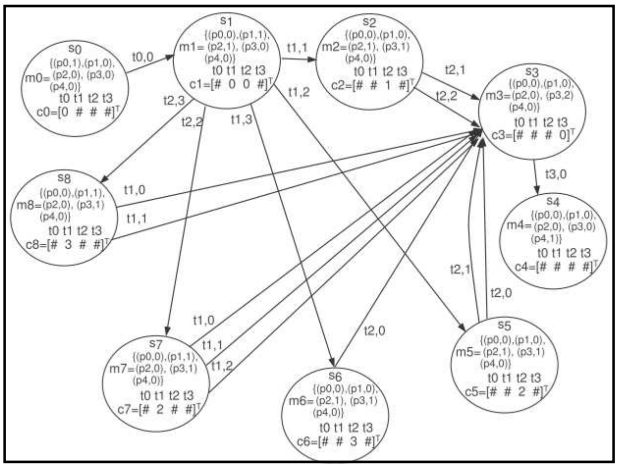


Time Petri Nets

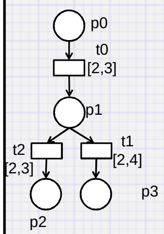


Time Petri Nets

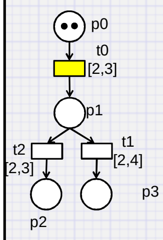




Time Petri Nets



Time Petri Nets



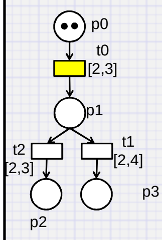
ET(s0)={t0}
c_{s0} (t0)=0

$I_D s_0$ (t0)=[2,3]

FDs_0(t0)=[2,3]

s0
m0={(p0,2),(p1,0),(p2,0),(p3,0),(p4,0)}
c0=[0 # #]^T

Time Petri Nets



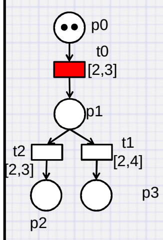
ET(s0)={t0}
c_{s0} (t0)=1

$I_D s_0$ (t0)=[1,2]

FDs_0(t0)=[2,3]

s0
m0={(p0,2),(p1,0),(p2,0),(p3,0),(p4,0)}
c0=[0 # #]^T

Time Petri Nets



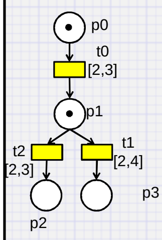
FT(s0)={t0}
c_{s0} (t0)=2

$I_D s_0$ (t0)=[0,1]

FDs_0(t0)=[2,3]

s0
m0={(p0,2),(p1,0),(p2,0),(p3,0),(p4,0)}
c0=[0 # #]^T

Time Petri Nets



ET(s1)={t0,t2,t1}
c_{s1} (t0)=0
c_{s1} (t1)=0
c_{s1} (t2)=0

$I_D s_1$ (t0)=[2,3]
$I_D s_1$ (t1)=[2,4]
$I_D s_1$ (t2)=[2,3]

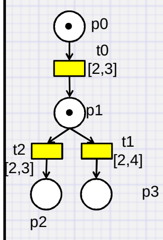
FDs_1(t0)=[2,3]
FDs_1(t1)=[2,3]
FDs_1(t2)=[2,3]

s0
m0={(p0,2),(p1,0),(p2,0),(p3,0),(p4,0)}
c0=[0 # #]^T

$\downarrow$ t0,2

s1
m1={(p0,1),(p1,1),(p2,0),(p3,0)}
c1=[0 0 0]^T

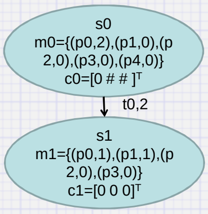
Time Petri Nets



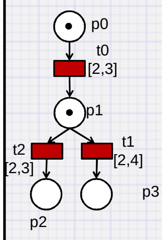
ET(s1)={t0,t2,t1}
c_{s1} (t0)=1
c_{s1} (t1)=1
c_{s1} (t2)=1

$I_D s_1$ (t0)=[1,2]
$I_D s_1$ (t1)=[1,3]
$I_D s_1$ (t2)=[1,2]

FDs_1(t0)=[2,3]
FDs_1(t1)=[2,3]
FDs_1(t2)=[2,3]



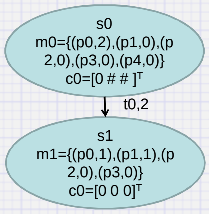
Time Petri Nets



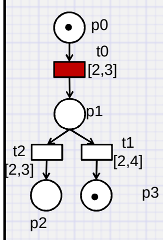
FT(s1)={t0,t2,t1}
c_{s1} (t0)=2
c_{s1} (t1)=2
c_{s1} (t2)=2

$I_D s_1$ (t0)=[0,1]
$I_D s_1$ (t1)=[0,2]
$I_D s_1$ (t2)=[0,1]

FDs_1(t0)=[2,3]
FDs_1(t1)=[2,3]
FDs_1(t2)=[2,3]



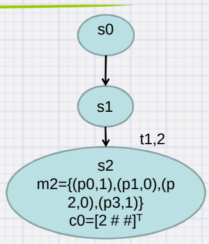
Time Petri Nets



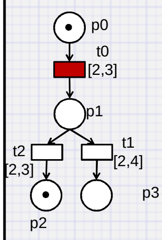
FT(s2)={t0}
c_{s2}(t0)=2

$I_D s_2$(t0)=[0,1]

FDs_2(t0)=[0,1]



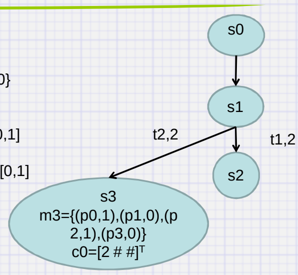
Time Petri Nets



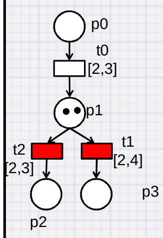
FT(s3)={t0}
c_{s3}(t0)=2

$I_D s_3$(t0)=[0,1]

FDs_3(t0)=[0,1]



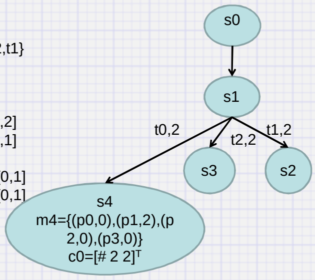
Time Petri Nets



FT(s1)={t2,t1}
c_{s4}(t1)=2
c_{s4}(t2)=2

$I_D s_4$(t1)=[0,2]
$I_D s_4$(t2)=[0,1]

FDs_4(t1)=[0,1]
FDs_4(t2)=[0,1]



Timed Petri Nets

Ramchandani74 e Zuberek87

Disparo em três fases. Duração. “Transição em disparo”

Infinite-server semantics

Veremos Zuberek87 (adota semântica de Passos)

Timed Petri Nets

Inp(p) = •p, Out(p)= p•, Inp(t)= •t, Out(t)=t•
Inh(t) = O conjunto de lugares inibidores de t

Um lugar p é free-choice, se, e somente se,
$\forall$ti,tj $\in$ Out(p): Inp(ti) = Inp(tj) $\wedge$inh(ti)=inh(tj).

Um lugar é guardado (*guarded)* se, e somente se,
$\forall$ti,tj $\in$ Out(p), $\exists$pk $\in$ P: pk $\in$ Inp(ti) $\wedge$ pk $\in$ Inh(tj)
$\vee$ pk $\in$ Inp(tj) $\wedge$ pk $\in$ Inh(ti)

Timed Petri Nets

$\mathbb{T}$ = (P,T,A,w,m0,c,f) , Timed Petri net

- P – Conjunto de lugares
- T – Conjunto de transições
- A $\subseteq$ (P x T) $\cup$ (T x P), Conjunto de arcos
- w:A$\rightarrow\aleph$,Peso dos arcos
- m0:P $\rightarrow\aleph$, marcação inicial

Free-choice Petri net: cada lugar é free-choice ou guarded
Partição de T em diferentes classes: Free(T) ={T1,T2,...,Tk}

- c:T$\rightarrow$0$\leq\Re\leq$1, função de probabilidade de escolha, tal que

$$\forall(T_i \in Free(T)) \sum_{t \in T_i} c(t) = 1$$

Timed Petri Nets

$\mathbb{T}$ = (P,T,A,w,m0,c,f) , Timed Petri net

- f: T $\rightarrow \Re^{+} \cup \{0\}$ – Duração

$\boxed{t_k \in En(m_i)}$

$$\forall(p \in P)\ m_j(p) = \begin{cases} m_i(p) - w(p,t_k), & \text{if } p \in Inp(t_k) - Out(t_k), \\ m_i(p) + w(t_k,p), & \text{if } p \in Out(t_k) - Inp(t_k), \\ m_i(p) - w(p,t_k) + w(t_k,p), & \text{if } p \in Inp(t_k) \cap Out(t_k), \\ m_i(p), & \text{otherwise.} \end{cases}$$

Timed Petri Nets

A selection function of a marking m in a net $\mathbf{N}$ is any function $g : T \rightarrow \{0, 1, ...\}$ such that:

- there exists a sequence of intermediate markings $(m_{i_1} m_{i_2} ... m_{i_k})$ and a corresponding sequence of transitions $\sigma = (t_{i_1} t_{i_2} ... t_{i_k})$ such that $m = m_{i_0}$, and $t_{i_j} \in En(m_{i_{j-1}})$ for all $1 \leq j \leq k$, where

$$\forall(p \in P)\ m_{i_j}(p) = m_{i_{j-1}}(p) - \begin{cases} w(p, t_{i_j}), & \text{if } p \in Inp(t_{i_j}), \\ 0, & \text{otherwise;} \end{cases}$$

- the set of transitions enabled by m_{i_k} is empty, and
- for all $t \in T$, $g(t)$ is equal to the number of occurrences of t in the sequence σ

The set of all selection functions of a marking m is denoted by $Sel(m)$.

Timed Petri Nets

s=(m,n,r) é um estado de uma TPN $\mathbb{T}$:

- m:P $\rightarrow \aleph$, é uma função de marcação
- n:T $\rightarrow \aleph$, *firing-ranking function* – função que indica o número de vezes que uma transição dispara naquele estado
- r(ti): $(\Re^{+} \cup \{0\})^{|k|}$, vetor que associa a cada disparo de t_i um numero real que representa *remaining firing time* disparo de t_i naquele estado. K é o número de vezes que t_i está sendo disparada em s (i.e., $n(t_i)=k$). Os valores do vetor são crescentes: r(ti)[1]<r(t)[2]<...<r(t)[k].

Timed Petri Nets

$s_i=(m_i,n_i,r_i)$ é o estado inicial (pode haver vários para uma free-choice net)

Escolhendo $n_i \in$ Sel(m0)

$$\forall(t \in T)\ r_i(t)[k] = \begin{cases} f(t), & \text{if } n_i(t) > 0 \wedge 1 \leq k \leq n_i(t), \\ \text{undefined}, & \text{otherwise;} \end{cases}$$

$$\forall(p \in P)\ m_i(p) = m_0(p) - \sum_{t \in Out(p)} w(p,t) * n_i(t).$$

Timed Petri Nets

$s_j=(m_j,n_j,r_j)$ é diretamente alcançado por $s_i=(m_i,n_i,r_i)$, satisfazendo as seguintes condições:

1. $h_i \; = \min_{t \in T \wedge n_i(t) > 0} (r_i(t)[1])$
2. $\forall (t \in T) \; d_i(t) \; = \begin{cases} z, & \text{if } n_i(t) \geq z \wedge \forall (1 \leq \ell \leq z) \; r_i(t)[\ell] = h_i, \\ 0, & \text{otherwise;} \end{cases}$
3. $\forall (p \in P) \; m'_i(p) \; = \; m_i(p) \; + \sum_{t \in Inp(p)} w(t,p) * d_i(t)$
4. $g_k \in Sel(m'_i)$
5. $\forall (p \in P) \; m_j(p) \; = \; m'_i(p) \; - \sum_{t \in Out(p)} w(p,t) * g_k(t)$
6. $\forall (t \in T) \; n_j(t) \; = \; n_i(t) \; - \; d_i(t) \; + \; g_k(t)$
7. $\forall (t \in T) \; r_j(t)[\ell] \; = \begin{cases} r_i(t)[\ell + d_i(t)] - h_i, & \text{if } 1 \leq \ell \leq n_i(t) - d_i(t), \\ f(t), & \text{if } n_i(t) - d_i(t) < \ell \leq n_j(t) \end{cases}$

Timed Petri Nets

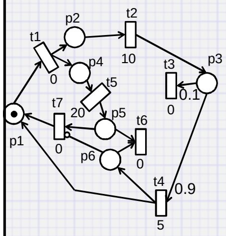
Grafo de alcançabilidade G=(V,D,h,q) de uma TPN $\mathbf{T}$

- V é conjunto de vértices, V=S(T) (conjunto de estados de T)
- D é o conjunto dos arcos dirigidos, $D \subset V \times V$. $(si,sj) \in D$, se, e somente se, é diretamente alcançável por si.
- Associa o *holding time* a cada estado

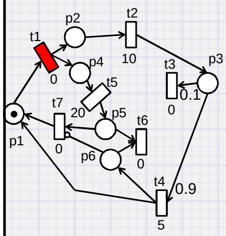
$$h(s_i) \; = \min_{t \in T \wedge n_i(t) > 0} (r_i(t)[1])$$

- q:D→[0,1] é uma função que associa uma probabilidade aos arcos do grafo

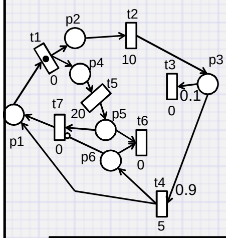
Exemplo



Exemplo



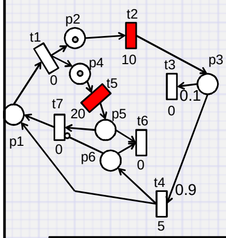
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1				

Marcações intermediárias:
m1'(p2)=1
m1'(p4)=1

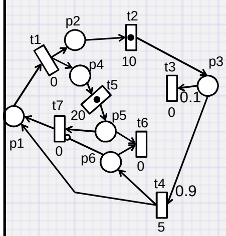
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1

Marcações intermediárias:
m1'(p2)=1
m1'(p4)=1

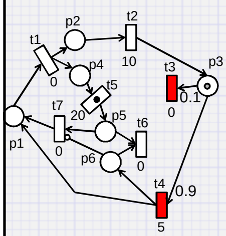
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10			

r_{s2}(t2)[1]=10
r_{s2}(t5)[1]=20

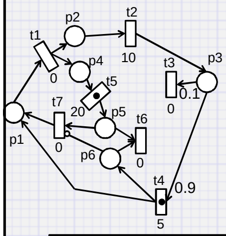
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1

r_{s2}(t2)[1]=10
r_{s2}(t5)[1]=20 (subtrair 10)
Marcações intermediárias:
m2'(p3)=1

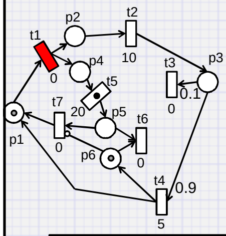
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1

$r_{s3}(t4)[1]=5$
$r_{s3}(t5)[1]=10$

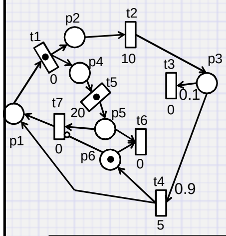
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1

$r_{s3}(t4)[1]=5$
$r_{s3}(t5)[1]=10$ (subtrair 5)
Marcações intermediárias:
m2'(p1)=1
m2'(p6)=1

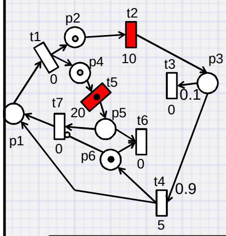
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0			

$r_{s5}(t1)[1]=0$
$r_{s5}(t5)[1]=5$

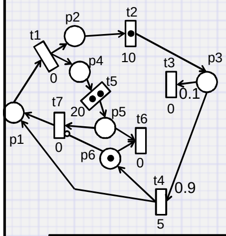
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1

$r_{s5}(t1)[1]=0$
$r_{s5}(t5)[1]=5$ (subtrair 0)
Marcações intermediárias:
m2'(p2)=1
m2'(p4)=1

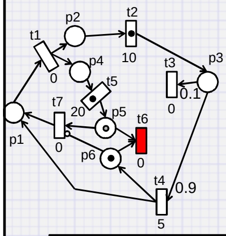
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5			

r_{s7}(t2)[1]=10
r_{s7}(t5)[1]=5
r_{s7}(t5)[2]=20

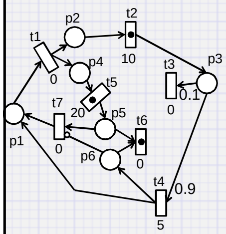
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1

r_{s7}(t2)[1]=10 (subtrair 5)
r_{s7}(t5)[1]=5
r_{s7}(t5)[2]=20 (subtrair 5)
Marcações Intermediárias:
m2'(p5)=1
m2'(p6)=1

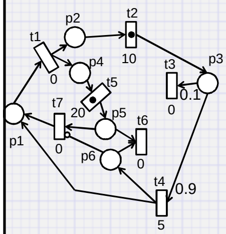
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1
8	todos 0	t7=1	0	g1=1	1	1
9	todos 0	t2=t5=t6=1	0	-	10	1

r_{s9}(t2)[1]=5
r_{s9}(t5)[1]=15
r_{s9}(t6)[1]=0

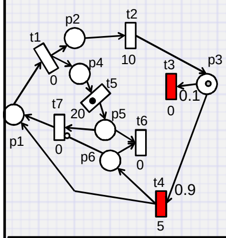
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1
8	todos 0	t7=1	0	g1=1	1	1
9	todos 0	t2=t5=t6=1	0	-	10	1
10	todos 0	t2=1,t5=1	5			

r_{s10}(t2)[1]=5
r_{s10}(t5)[1]=15

Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1
8	todos 0	t7=1	0	g1=1	1	1
9	todos 0	t2=t5=t6=1	0	-	10	1
10	todos 0	t2=1,t5=1	5	t4=1	3	0.9
				t3=1	4	0.1

$r_{s10}(t2)[1]=5$

$r_{s10}(t5)[1]=15$ (subtrair 5)

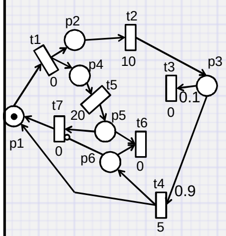
Marcações Intermediárias:

m10'(p3)=1

Tempo de Execução

- Análise do grafo de estados + Algoritmo de procura de caminhos (Redes Genéricas)
- Métodos estruturais

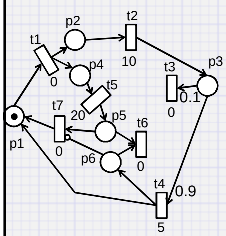
Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1
8	todos 0	t7=1	0	g1=1	1	1
9	todos 0	t2=t5=t6=1	0	-	10	1
10	todos 0	t2=1,t5=1	5	t4=1	3	0.9
				t3=1	4	0.1

Qual é o menor caminho s1 até s5?

Exemplo



si	mi	ni	h(si)	gk	sj	q(si,sj)
1	todos 0	t1=1	0	t2=1,t5=1	2	1
2	todos 0	t2=1,t5=1	10	t4=1	3	0.9
				t3=1	4	0.1
3	todos 0	t4=1,t5=1	5	t1=1	5	1
4	todos 0	t3=1,t5=1	0	-	6	1
5	p6=1	t1=1,t5=1	0	t2=1,t5=1	7	1
6	todos 0	t5,=1	10	t7=1	8	1
7	p6=1	t2=1,t5=2	5	g6=1	9	1
8	todos 0	t7=1	0	g1=1	1	1
9	todos 0	t2=t5=t6=1	0	-	10	1
10	todos 0	t2=1,t5=1	5	t4=1	3	0.9
				t3=1	4	0.1

D(s1,s5)=0 + 10 + 5 + 0= 15

Timed Petri Nets - INA

$T = (P,T,F,W,m0,D)$, Timed Petri net

- P – Conjunto de lugares
- T – Conjunto de transições
- $F \subseteq (P \times T) \cup (T \times P)$, Conjunto de arcos
- $W: A \rightarrow \aleph$, Peso dos arcos
- $D: T \rightarrow \aleph$, Duração da transição

Adota semântica de passos com single-server firing semantics

Timed Petri Nets - INA

$S \subseteq (M,A)$ conjunto de todos os estados, onde

- $M \subseteq (P \times \aleph)$: conjunto de marcações
- $A \subseteq (T \times \aleph)$: conjunto das durações (tempo) restantes de disparo das transições

Um estado $s \in S$ é uma tupla $s = (m,a)$, no qual *m* $\in M$ é a marcação e *a* $\in A$ a duração restante das transições em disparo.

Se $a(t) = 0$, a transição t não está disparando no estado s

$s0 = (m0,\underline{0})$ é a marcação inicial. $\underline{0}(t) = 0, \forall t \in T$

Timed Petri Nets - INA

$U \subseteq T$ é um passo máximo no estado $s=(m,a)$, se e somente se:

- $\forall t \in U, \quad a(t) = 0$;
- $\forall p \in P$, $m(p) >= \sum_{t \in U} W(p,t)$
- $U = \{\}$**: (i)**$\forall t \in ET(m), a(t) >= 0$**;** ou **(ii)**$\forall t \in T, ET(m) = \{\}$ e $a(t) >= 0$**,**
- $\nexists U'$ satisfazendo as condições acima, tal que $U \subset U'$

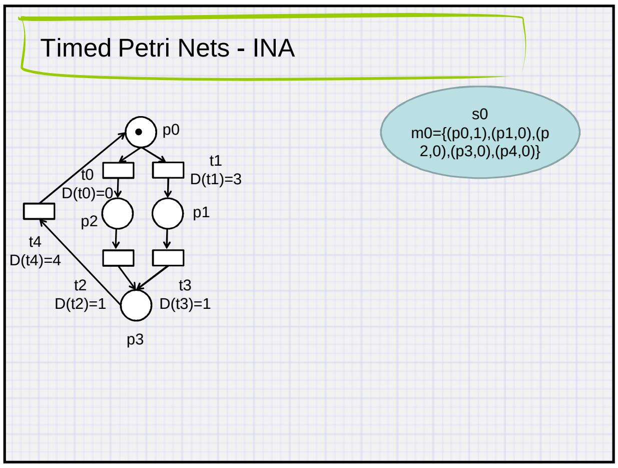
Conjunto de transições habilitadas:

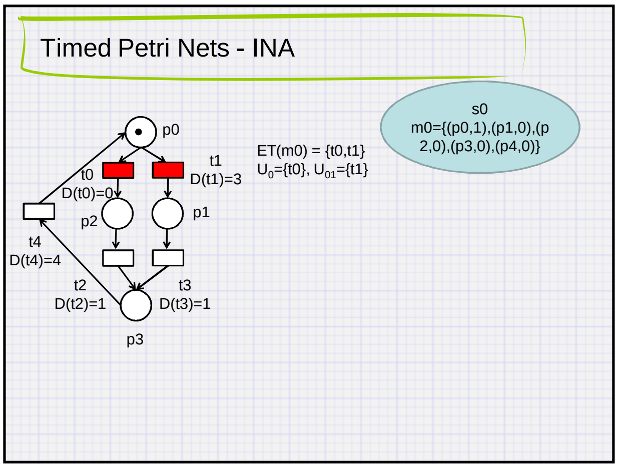
$ET(m) = \{t \mid m(p) >= W(p,t)\}, \forall p \in P$

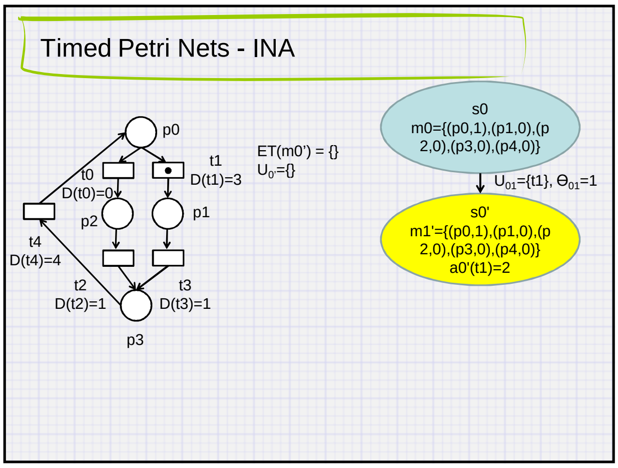
Timed Petri Nets - INA

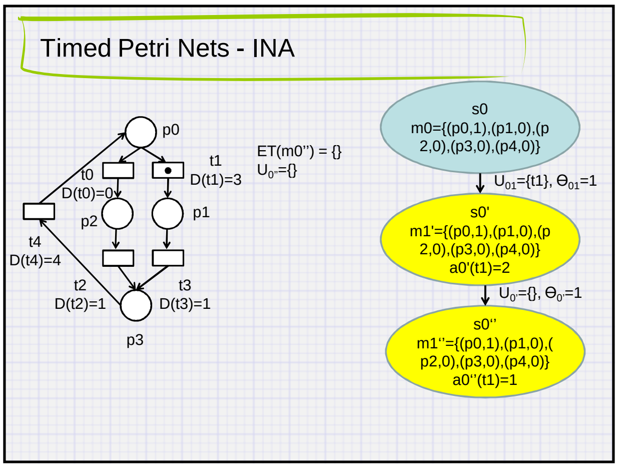
Assuma o estado $s=(m,a)$ e U um passo máximo em s. O estado $s'=(m',a')$ é alcançado devido ao disparo de U em s, da seguinte forma:

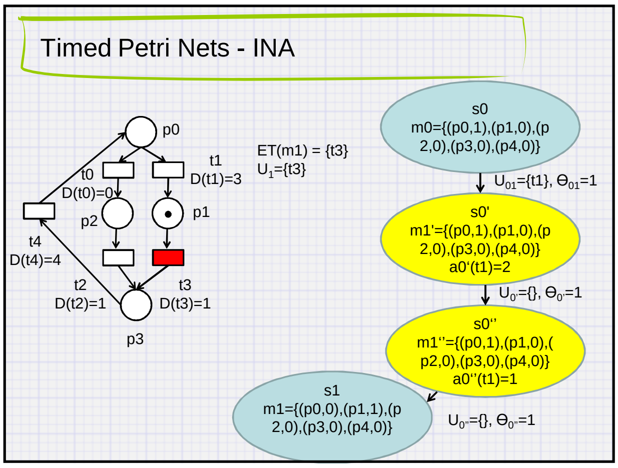
- $\Theta = \min(1,D(t)), \forall t \in U$
- $m'(p) = m(p) - \sum_{t \in U} W(p,t) + \sum_{t \in U \wedge D(t)=\Theta} W(t,p) + \sum_{a(t)>0 \wedge a(t)=\Theta} W(t,p), \forall p \in P$
- $a'(t) = \begin{cases} D(t) - \Theta, \text{ if } t \in U \\ a(t) - \Theta, \text{ if } t \notin U \wedge a(t) > 0 \\ 0, \text{ caso contrário} \end{cases}$

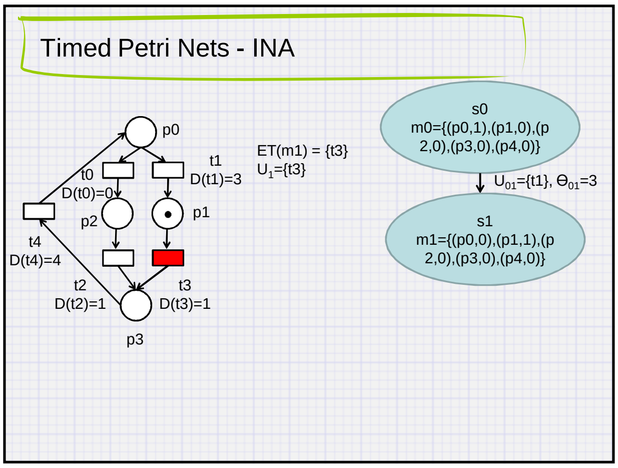

Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}

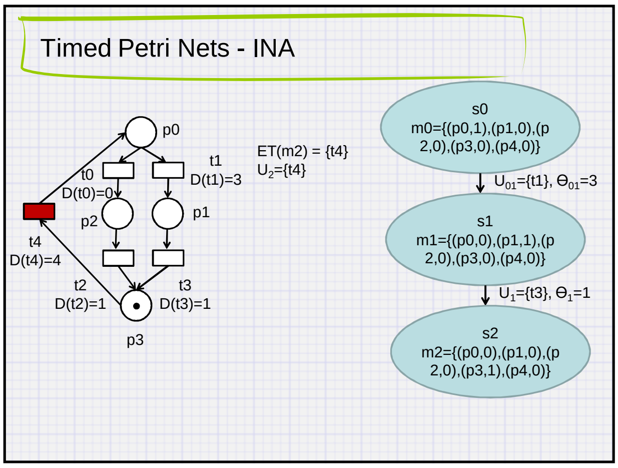

Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
ET(m0) = {t0,t1}
U_0={t0}, U_{01}={t1}
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}

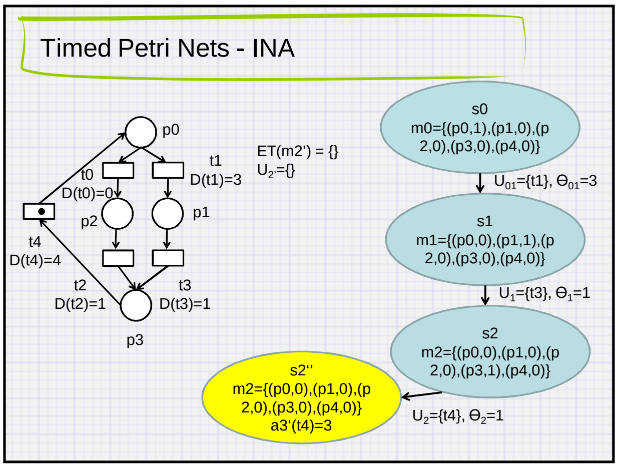

Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
ET(m0') = {}
U_0={}
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U_{01}={t1}, Θ_{01}=1
s0'
m1'={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
a0'(t1)=2


Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
ET(m0'') = {}
U_0={}
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U_{01}={t1}, Θ_{01}=1
s0'
m1'={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
a0'(t1)=2
U_0={}, Θ_0=1
s0''
m1''={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
a0''(t1)=1

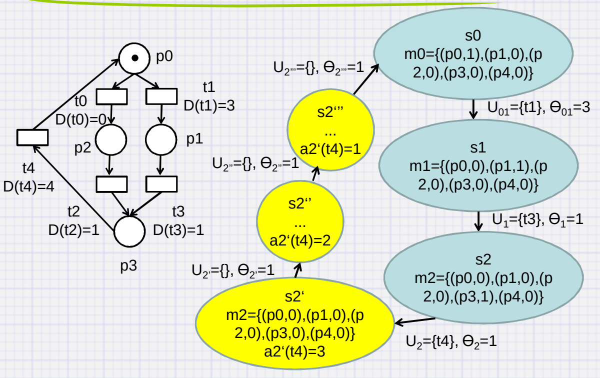

Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
ET(m1) = {t3}
U1={t3}
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U01={t1}, Θ01=1
s0'
m1'={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
a0'(t1)=2
U0'={}, Θ0'=1
s0''
m1''={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
a0''(t1)=1
s1
m1={(p0,0),(p1,1),(p2,0),(p3,0),(p4,0)}
U0''={}, Θ0''=1


Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
ET(m1) = {t3}
U1={t3}
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U01={t1}, Θ01=3
s1
m1={(p0,0),(p1,1),(p2,0),(p3,0),(p4,0)}

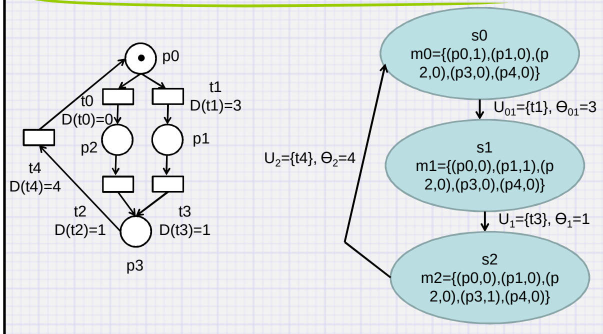

Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
ET(m2) = {t4}
U2={t4}
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U01={t1}, Θ01=3
s1
m1={(p0,0),(p1,1),(p2,0),(p3,0),(p4,0)}
U1={t3}, Θ1=1
s2
m2={(p0,0),(p1,0),(p2,0),(p3,1),(p4,0)}


Timed Petri Nets - INA
p0
t0
D(t0)=0
t1
D(t1)=3
p2
p1
t4
D(t4)=4
t2
D(t2)=1
t3
D(t3)=1
p3
ET(m2') = {}
U2={}
s0
m0={(p0,1),(p1,0),(p2,0),(p3,0),(p4,0)}
U01={t1}, Θ01=3
s1
m1={(p0,0),(p1,1),(p2,0),(p3,0),(p4,0)}
U1={t3}, Θ1=1
s2
m2={(p0,0),(p1,0),(p2,0),(p3,1),(p4,0)}
s2''
m2={(p0,0),(p1,0),(p2,0),(p3,0),(p4,0)}
a3'(t4)=3
U2={t4}, Θ2=1

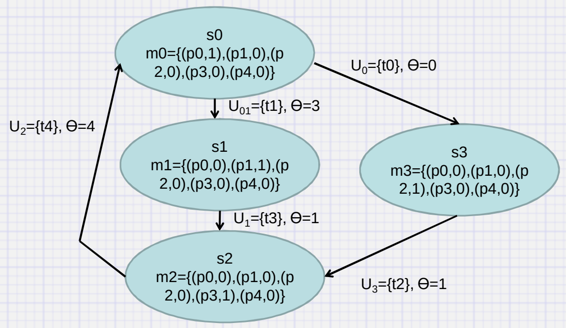
Timed Petri Nets - INA



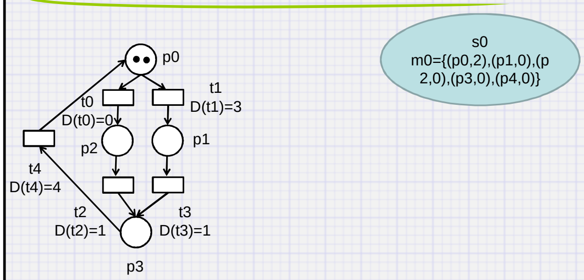
Timed Petri Nets - INA



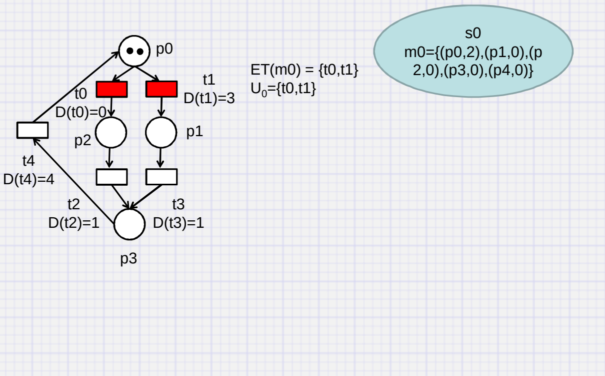
Timed Petri Nets - INA



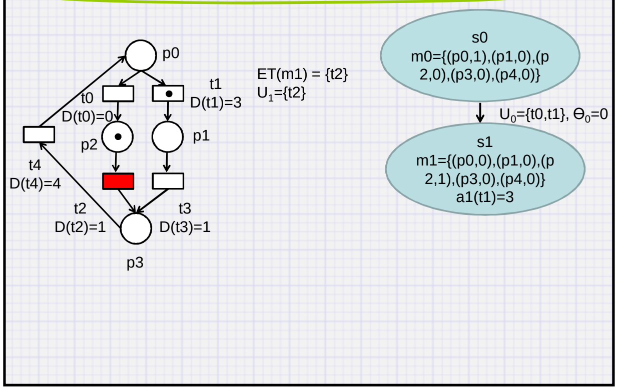
Timed Petri Nets - INA



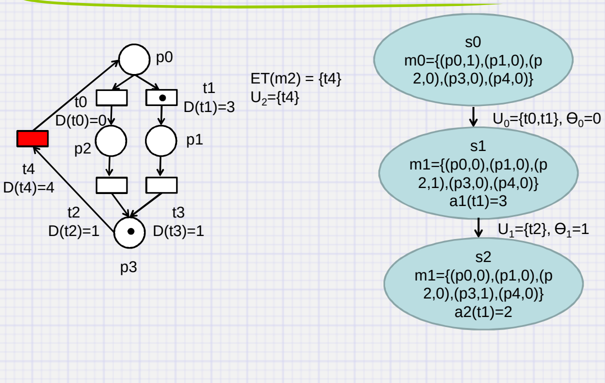
Timed Petri Nets - INA



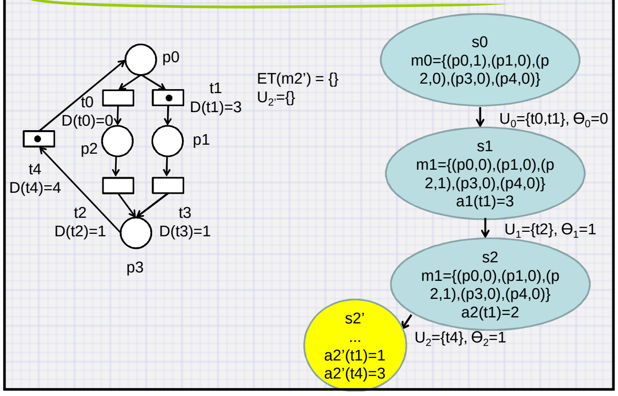
Timed Petri Nets - INA



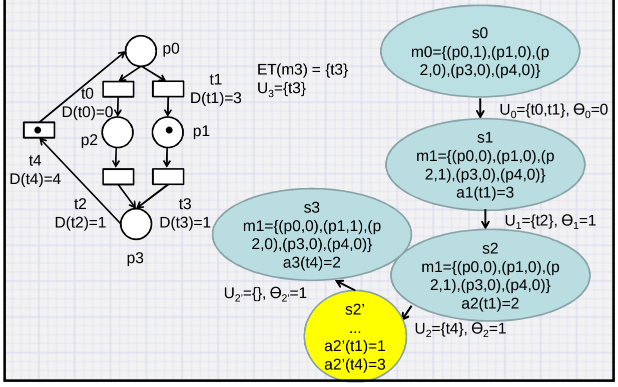
Timed Petri Nets - INA



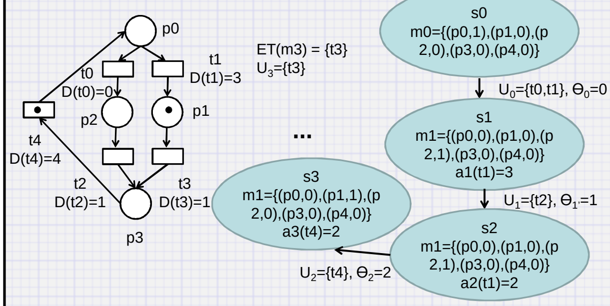
Timed Petri Nets - INA



Timed Petri Nets - INA

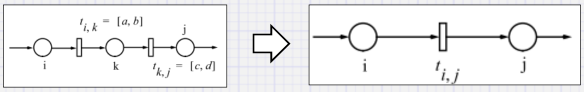


Timed Petri Nets - INA



Regras de Redução

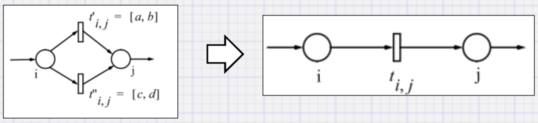
Técnicas de reduções para uma rede sequencial



$$t_{i,j} = [a + c, b + d]$$

Regras de Redução

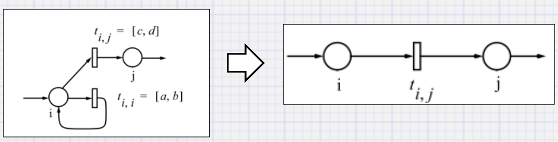
Técnicas de reduções para uma rede sequencial



$$t_{i,j} = [min(a, c), max(b, d)]$$

Regras de Redução

Técnicas de reduções para uma rede sequencial

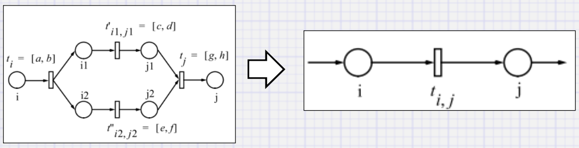


$t_{i,j} = [ma + c, nb + d]$

m - *lower bound*, n - *upper bound* } *of iterations*

Regras de Redução

Técnicas de reduções para uma rede sequencial



$$t_{i,j} = [\,a + max(c, e) + g,\; b + max(d, f) + h\,]$$

Leitura

F. D. J. Bowden. A Brief Survey and Synthesis of the Roles of Time in Petri Nets. *Mathematical and Computter Modelling,* 2000.

E. Tavares. Software Synthesis for Energy-Constrained Hard Real-Time Systems, 2009.

W. Zuberek. Timed Petri Nets: Definitions, Properties and Applications. Microelectronics and Reliability, 1991

Zeugmann and et al. Worst-case Analysis of Concurrent Systems with Duration Interval Petri Nets. Informatik-Bericht,1997.

Leitura

B. Berthomieu and M. Diaz. Modeling and Verification of Time Dependent Systems Using Time Petri Nets. IEEE Trans. Software Engineering, 1991.

N. Leveson e J. Stolzy. Safety Analysis Using Petri Nets. IEEE Trans. Software Engineering, 1987.

Exemplo

- Tavares09 e Barreto05
- Síntese de software para sistemas de tempo real crítico com restrições de energia
- Geração de código sob medida
- Escalonador Híbrido
- Adoção de DVS

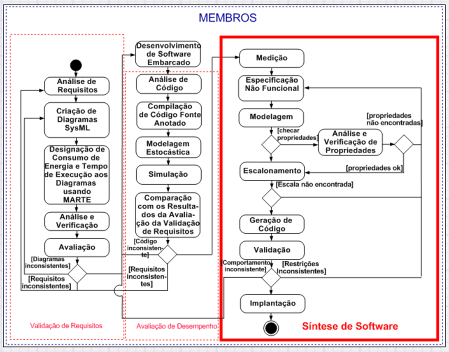
```
void codeT1() {...}
void codeT2() {...}
#define SCHEDULE_SIZE 5
struct SchItem sch[SCHEDULE_SIZE] =
{
  {0, INSTANCE, 1, 2V/20MHz,(int *)codeT1},
  {5, INSTANCE, 2, 2V/20MHz,(int *)codeT2},
  {7, VOLT_SWITCH, 2, 1V/10MHz,(int *)codeT2},
  {9, RETURN, 1, 2V/20MHz,(int *)codeT1},
  {12, VOLT_SWITCH, 1, 1V/10MHz, (int *)codeT1},
};
```

113

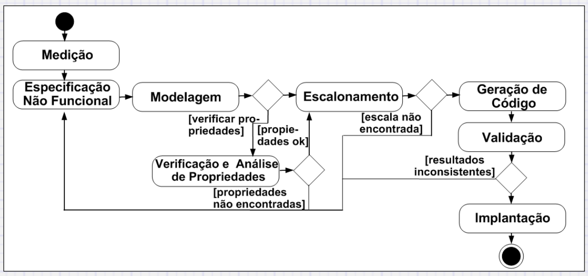
Exemplo

- Sistemas de Tempo Real Crítico
  - Execuções satisfazendo as restrições temporais
  - Restrições temporais críticas
  - Restrições de Energia e Temporais: **usualmente conflitantes**

Exemplo



Exemplo



Exemplo

Composta por tarefas concorrentes periódicas
Restrições Temporais

Tarefas Periódicas (**ph_p,r_p,c_p,d_p ,p_p ,$code_p$**)
- ph = fase
- r = *release*
- c = pior caso de ciclos de execução (WCEC)
- d = *deadline*
- p = período
- code= código

Tarefas Esporádicas (**c_s,d_s,min_s,$code_s$**)

min = menor período entre duas ativações

Tradução de tarefas esporádicas para periódicas

Exemplo

Relação entre Tarefas
- Precedência
- Exclusão

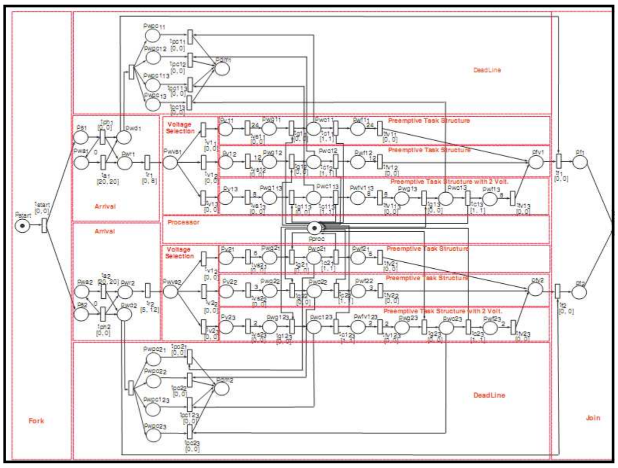
Método de Escalonamento (preempção ou não)

Adoção de uma unidade de tempo (*Task Time Unit* - TTU)

Informações sobre o despachante

Arquitetura do hardware
- DVS: Níveis de tensão (e as respectivas máximas frequências)
- Consumo de energia por ciclo em cada nível

Restrição de Energia do Sistema

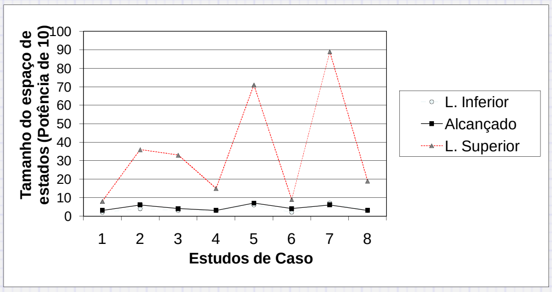


Exemplo

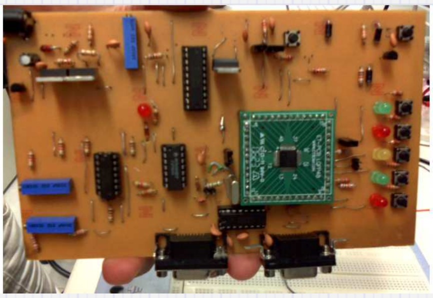
- Geração de escalas

Inst.	Size	Schedule	Found	W/DVS(J)	O/DVS(J)	%	Ipedf	Time(s)
4	7×10^{7}	48	141	0.24740	0.31320	21%	45%	0.001
6	7×10^{35}	4377	518406	0.00069	0.00105	34%	54%	35.200
12	2×10^{32}	551	9906	267.00000	360.00000	26%	29%	0.282
4	5×10^{14}	246	246	279.00000	371.00000	25%	25%	0.003
289	9×10^{70}	235852	1884381	0.11900	0.34500	66%	73%	291.221
10	2×10^{8}	83	4268	0.00021	0.00023	9%	39%	0.234
3604	3×10^{68}	381313	381313	3.86200	4.76600	19%	19%	9.606
10	9×10^{18}	320	85085	0.01607	0.01682	4%	27%	0.395

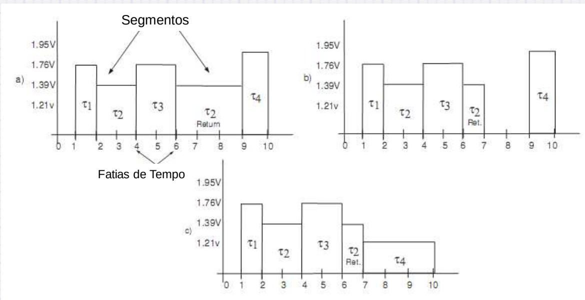
Exemplo



Exemplo



Escalonador Runtime (Abordagem Híbrida)



Exemplo