Time and Petri Nets - A Way for Modeling and Verification of Time-dependent Concurrent Systems

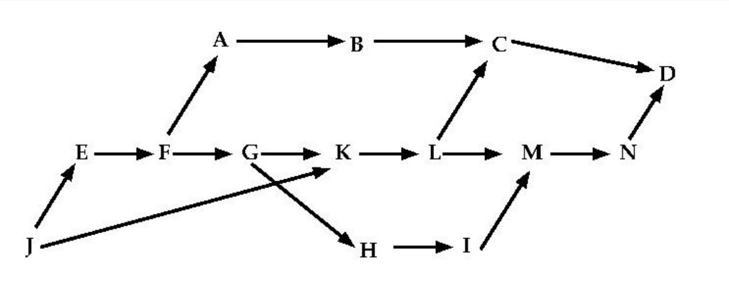
Louchka Popova-Zeugmann

Humboldt-Universität zu Berlin
Department of Computer Science

September 16, 2013
Moscow

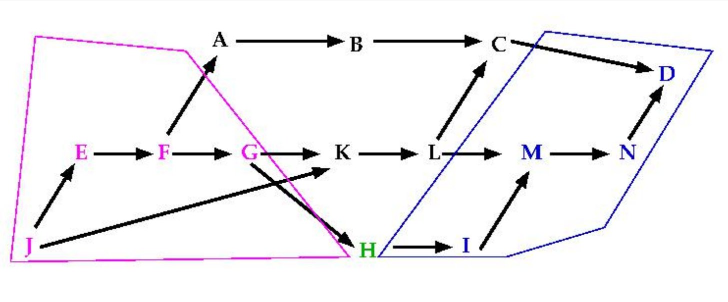
Outline

Concurrency



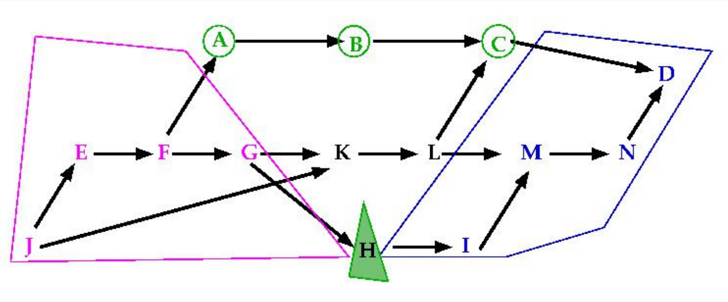
A, B, ... , N: events.

Concurrency



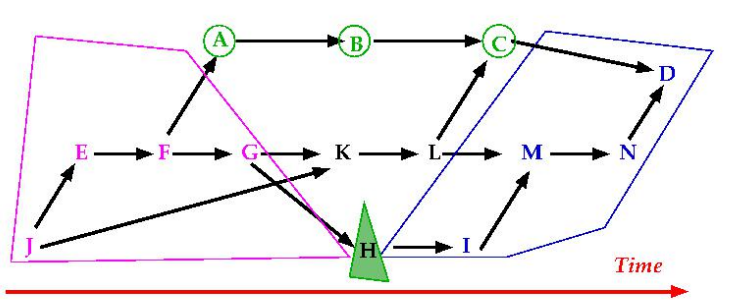
A, B, ... , N: events.

Concurrency



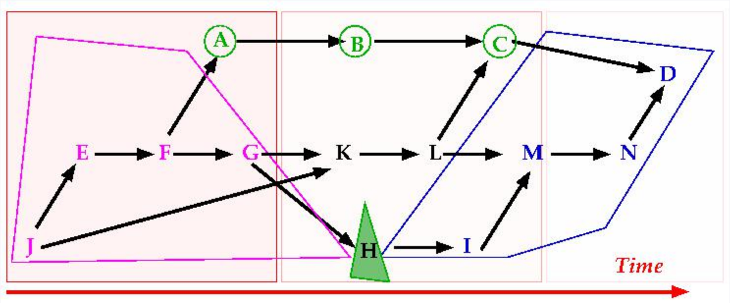
A, B, ... , N: events.

Concurrency and Time?



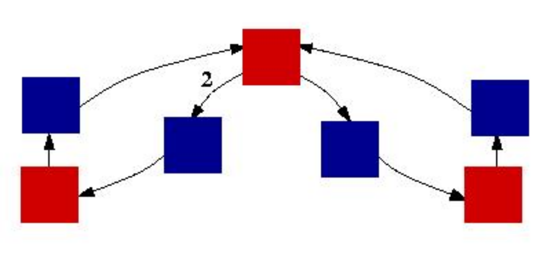
A, B, ... , N: events.

Concurrency and Time



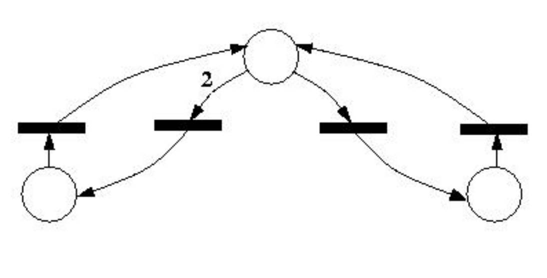
A, B, ... , N: events.

Statics: non initialized Petri Net



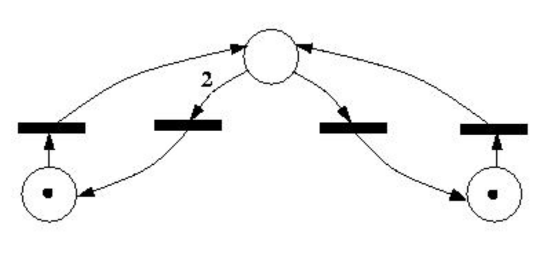
finite two-coloured weighted directed graph

Statics: non initialized Petri Net

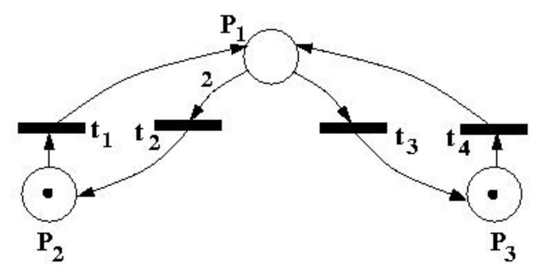


finite two-coloured weighted directed graph

Statics: initialized Petri Net

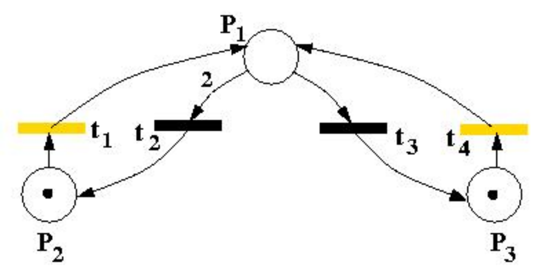


Statics: initialized Petri Net



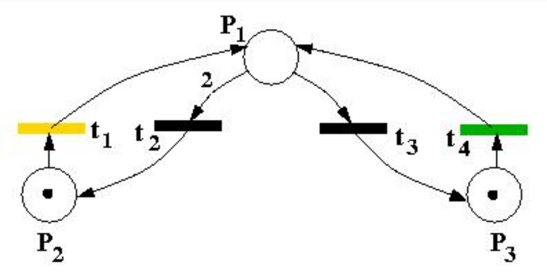
initial marking: $m_0 = (0, 1, 1)$

Dynamics: firing rule



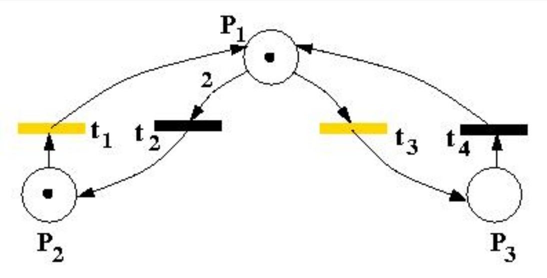
$$m_0 = (0, 1, 1)$$

Dynamics: firing rule



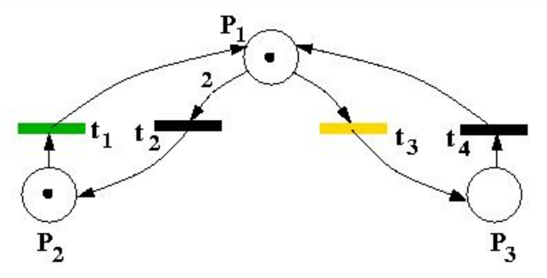
$$m_0 = (0, 1, 1)$$

Dynamics: firing rule



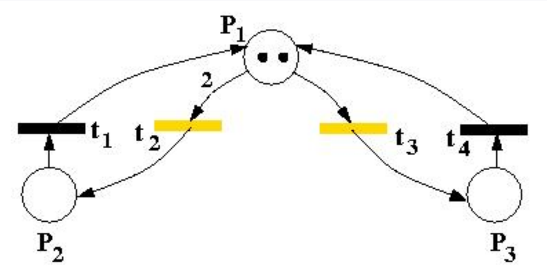
$$m_0 = (0,1,1)$$
$$m_1 = (1,1,0)$$

Dynamics: firing rule



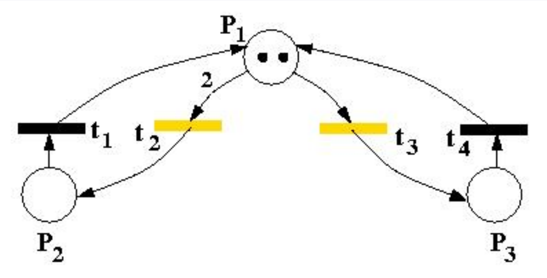
$$m_0 = (0, 1, 1)$$
$$m_1 = (1, 1, 0)$$

Dynamics: firing rule



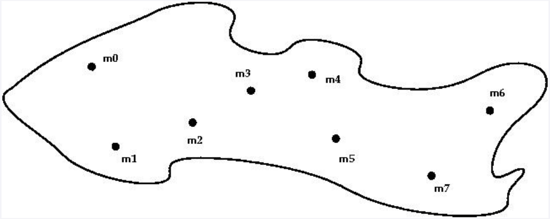
$$m_0 = (0, 1, 1)$$
$$m_1 = (1, 1, 0)$$
$$m_2 = (2, 0, 0)$$

Dynamics: firing rule



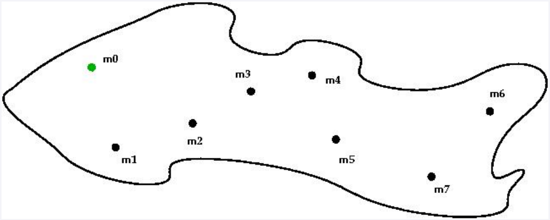
$$m_0 = (0,1,1)$$
$$m_1 = (1,1,0)$$
$$m_2 = (2,0,0)$$
$$\vdots$$

State Space



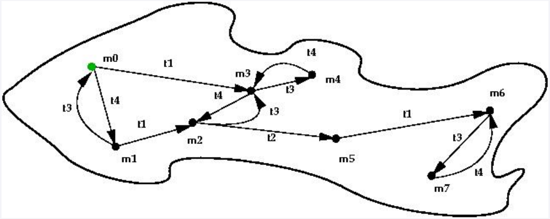
- The state space is the set of all reachable markings starting in m_0.

State Space



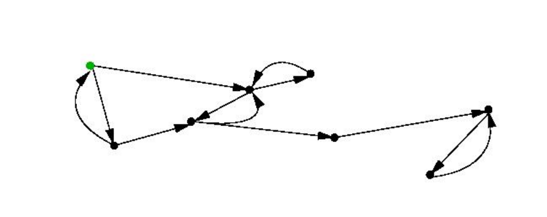
- The state space is the set of all reachable markings starting in m_0.

State Space

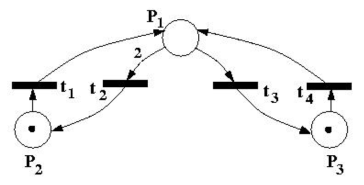


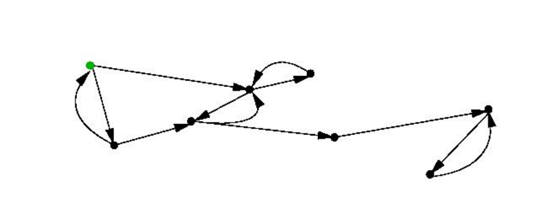
- The state space is the set of all reachable markings starting in m_0.
- All reachable markings + firing relation

State Space

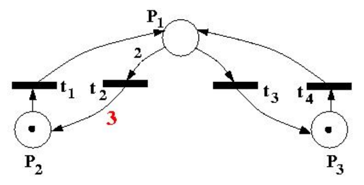


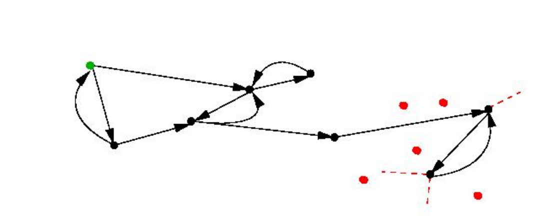
- The state space is the set of all reachable markings starting in m_0.
- All reachable markings + firing relation = reachability graph of the PN





The reachability graph is finite





The reachability graph is infinite

Petri Nets and Turing Machines

Remark:

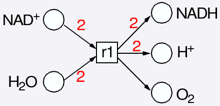
Classic Petri Nets are not Turing-complete.

Bionetworks, Basics

- **chemical reactions -> atomic actions -> Petri net transitions**

$$2\ NAD^+ + 2\ H_2O \rightarrow 2\ NADH + 2\ H^+ + O_2$$

input compounds

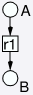


output compounds

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

r1: A -> B



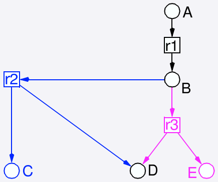
popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

Bionetworks, Intro

r1: A -> B

r2: B -> C + D

r3: B -> D + E



-> alternative reactions

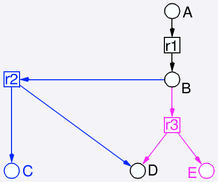
popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

r1: A -> B

r2: B -> C + D

r3: B -> D + E



-> alternative reactions

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

PN & Systems Biology

r1: A -> B

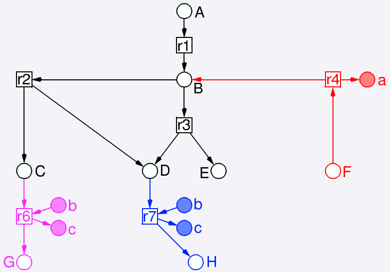
r2: B -> C + D

r3: B -> D + E

r4: F -> B + a

r6: C + b -> G + c

r7: D + b -> H + c



-> *concurrent reactions*

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

PN & Systems Biology

r1: A -> B

r2: B -> C + D

r3: B -> D + E

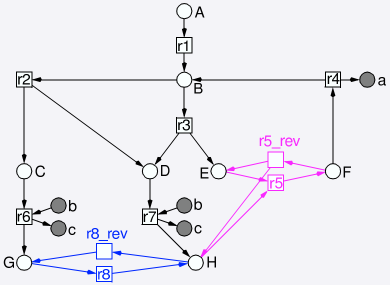
r4: F -> B + a

r5: E + H <-> F

r6: C + b -> G + c

r7: D + b -> H + c

r8: H <-> G



-> reversible reactions

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

PN & Systems Biology

r1: A -> B

r2: B -> C + D

r3: B -> D + E

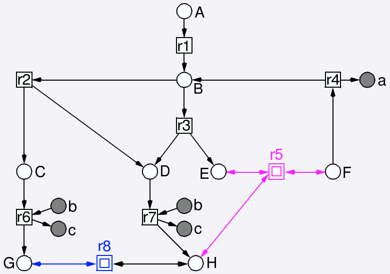
r4: F -> B + a

r5: E + H <-> F

r6: C + b -> G + c

r7: D + b -> H + c

r8: H <-> G



-> reversible reactions
- hierarchical nodes

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

r1: A -> B

r2: B -> C + D

r3: B -> D + E

r4: F -> B + a

r5: E + H <-> F

r6: C + b -> G + c

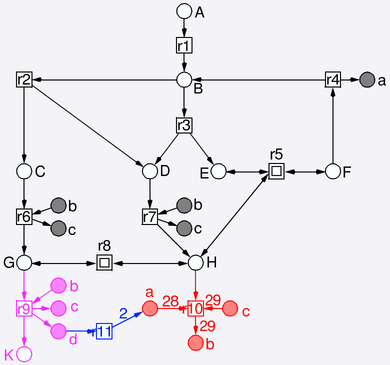
r7: D + b -> H + c

r8: H <-> G

r9: G + b -> K + c + d

r10: H + 28a + 29c -> 29b

r11: d -> 2a



popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

Bionetworks, Intro

r1: A -> B

r2: B -> C + D

r3: B -> D + E

r4: F -> B + a

r5: E + H <-> F

r6: C + b -> G + c

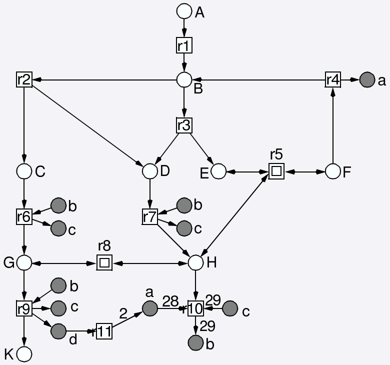
r7: D + b -> H + c

r8: H <-> G

r9: G + b -> K + c + d

r10: H + 28a + 29c -> 29b

r11: d -> 2a



popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

BIONETWORKS, INTRO

r1: A -> B

r2: B -> C + D

r3: B -> D + E

r4: F -> B + a

r5: E + H <-> F

r6: C + b -> G + c

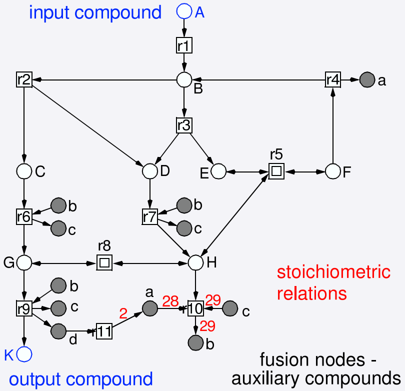
r7: D + b -> H + c

r8: H <-> G

r9: G + b -> K + c + d

r10: H + 28a + 29c -> 29b

r11: d -> 2a



popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

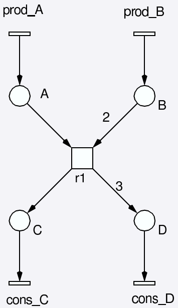
Aims of Analyzing in Metabolic Systems

Input: A metabolic system

Output:

- Is a *steady state* reachable there?
- Behavior of the metabolic system by *changing* the concentration of the substances.
- Behavior of the metabolic system by *knocking out* of enzymes.
- Timeframe for special series of reactions.
- Checking the consistence of the metabolic system, etc..

TRANSFORMATION, EX1



-> properties as time-less net

INA

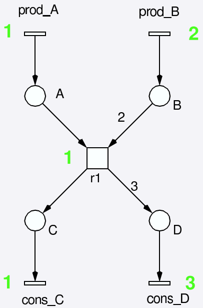
ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	N	N	Y	N	?	N	Y	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

TRANSFORMATION, EX1

PN & Systems Biology



T-INVARIANTE

-> properties as time-less net

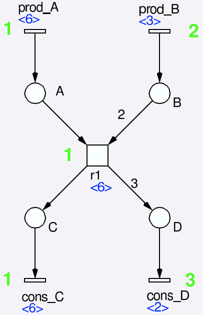
INA

ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	N	N	Y	N	?	N	Y	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

TRANSFORMATION, EX1



T-INVARIANTE

-> properties as time net

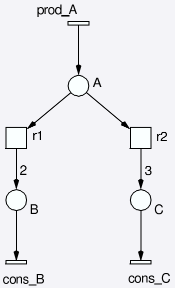
INA

ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	Y	N	N	N	?	N	Y	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

TRANSFORMATION, EX2



-> properties as time-less net

INA

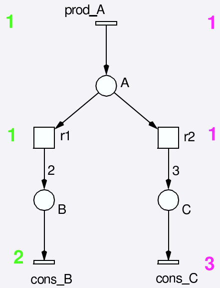
ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	N	N	Y	N	?	N	N	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

TRANSFORMATION, EX2

PN & Systems Biology



T-INVARIANTE1
T-INVARIANTE2

-> properties as time-less net

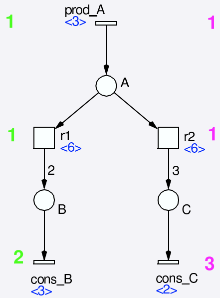
INA

ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	N	N	Y	N	?	N	N	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

TRANSFORMATION, EX2



T-INVARIANTE1

T-INVARIANTE2

-> properties as time net

INA

ORD	HOM	NBM	PUR	CSV	SCF	CON	SC	Ft0	tF0	Fp0	pF0	MG	SM	FC	EFC	ES
N	Y	N	Y	N	Y	Y	N	Y	Y	N	N	Y	N	Y	Y	Y

CPI	CTI	B	SB	REV	DSt	BSt	DTr	DCF	L	LV	L&S
N	Y	Y	N	N	N	?	N	Y	Y	Y	N

popova@informatik.hu-berlin.de, monika.heiner@informatik.tu-cottbus.de, ina.koch@tfh-berlin.de

Adequate Aims of Analyzing in the Model

Input: A Petri net

Output:

- Coverability by T-invariants?
- Coverability by T-invariants in a time-dependent extension of the Petri net.
- Behavior of the time-dependent Petri net by removing of subnets.
- Length of time of special firing sequences (quantitative analysis).
- Liveness, be deadlock-free, etc..(qualitative analysis)

Time Assignment

- time dependent Petri Nets with time specification at
  - transitions
  - places
  - arcs
  - tokens

Time Assignment

- time dependent Petri Nets with time specification at
  - transitions
  - places
  - arcs
  - tokens

- time dependent Petri Nets with
  - deterministic
  - stochastic

  time assignment.

Time Assignment

- time dependent Petri Nets with time specification at
  - transitions
  - places
  - arcs
  - tokens

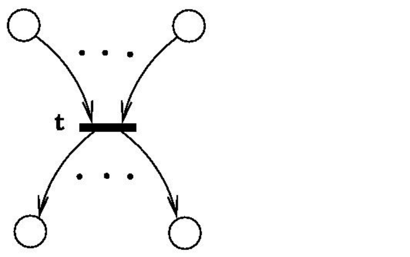
- time dependent Petri Nets with
  - deterministic
  - stochastic

  time assignment.

Time Petri nets

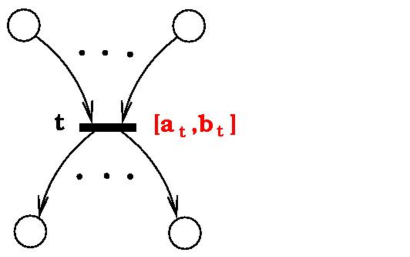
Time Petri nets

Definition (informal)



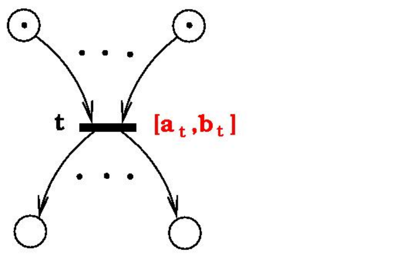
Time Petri nets

Definition (informal)



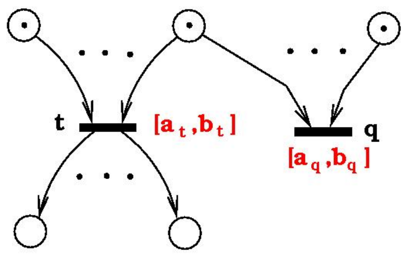
Time Petri nets

Definition (informal)



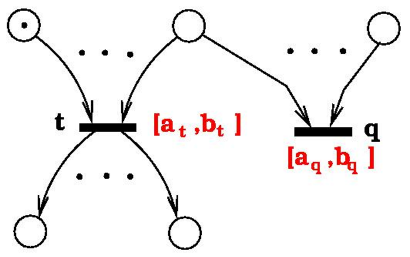
Time Petri nets

Definition (informal)

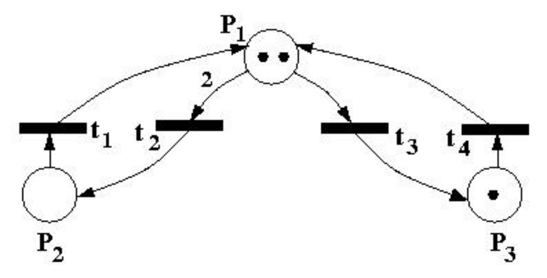


Time Petri nets

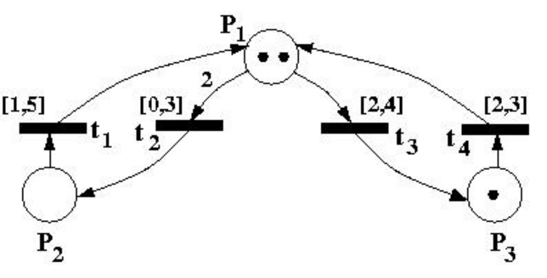
Definition (informal)



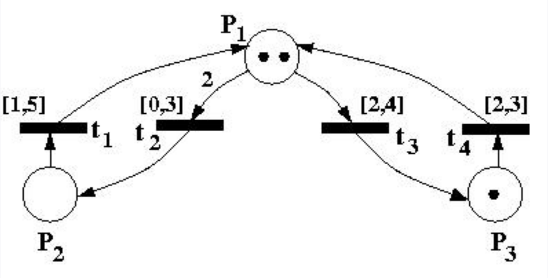
Statics: Petri Net (Skeleton)



Statics: **Time Petri Net**

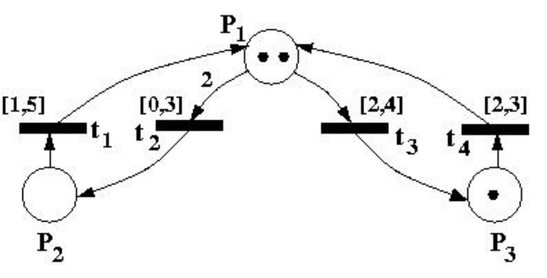


Statics: **Time Petri Net**



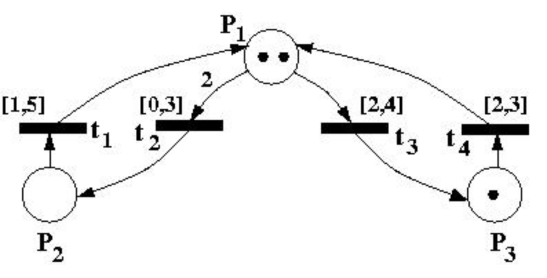
- $m_0 = (2, 0, 1)$

Statics: **Time Petri Net**



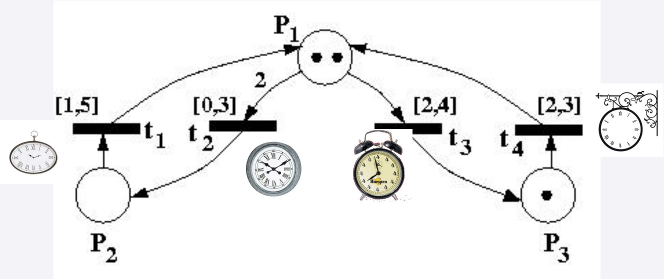
- $m_0 = (2, 0, 1)$ *p*-marking

Statics: **Time Petri Net**



- $m_0 = (2, 0, 1)$ *p*-marking
- $h_0 = (\sharp, 0, 0, 0)$ *t*-marking

Statics: **Time Petri Net**



- $m_0 = (2, 0, 1)$ *p*-marking
- $h_0 = (\sharp, 0, 0, 0)$ *t*-marking

$h(t)$ is the time shown by the clock of t since the last enabling of t

Time Petri nets

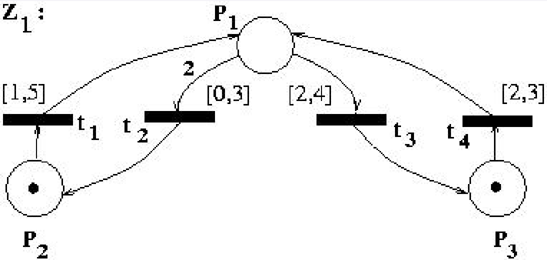
- $z = (m, h)$ state

Dynamics: state change

firing of a transition / \ **elapsing of time**

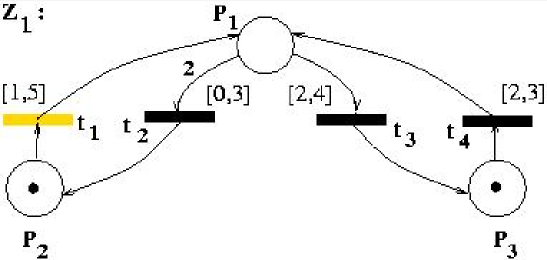
Notation: $z \xrightarrow{t} z'$ and $z \xrightarrow{\tau} z'$

An example



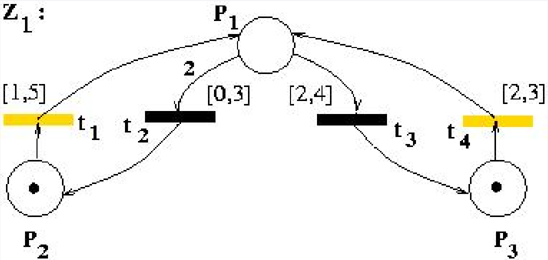
$$(m_0, \begin{pmatrix} 0 \\ \sharp \\ \sharp \\ 0 \end{pmatrix})$$

An example



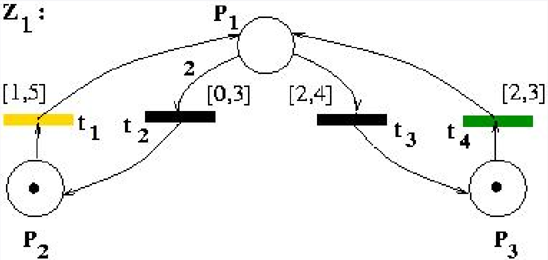
$$(m_0, \begin{pmatrix} 0 \\ \sharp \\ \sharp \\ 0 \end{pmatrix}) \xrightarrow{1.3} (m_1, \begin{pmatrix} 1.3 \\ \sharp \\ \sharp \\ 1.3 \end{pmatrix})$$

An example



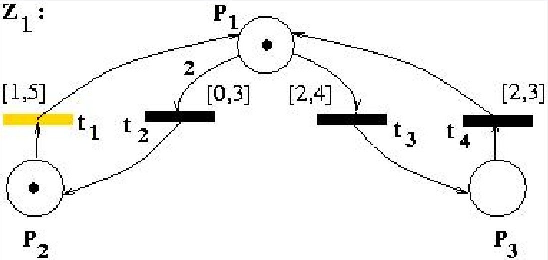
$$z_0 \xrightarrow{1.3} (m_1, \begin{pmatrix} 1.3 \\ \sharp \\ \sharp \\ 1.3 \end{pmatrix}) \xrightarrow{1.0} (m_2, \begin{pmatrix} 2.3 \\ \sharp \\ \sharp \\ 2.3 \end{pmatrix})$$

An example



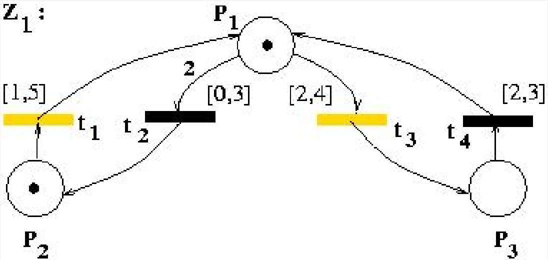
$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \left(m_2, \begin{pmatrix} 2.3 \\ \sharp \\ \sharp \\ 2.3 \end{pmatrix}\right) \xrightarrow{t_4}$$

An example



$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \left(m_2, \begin{pmatrix} 2.3 \\ \sharp \\ \sharp \\ 2.3 \end{pmatrix}\right) \xrightarrow{t_4} \left(m_3, \begin{pmatrix} 2.3 \\ \sharp \\ 0.0 \\ \sharp \end{pmatrix}\right)$$

An example



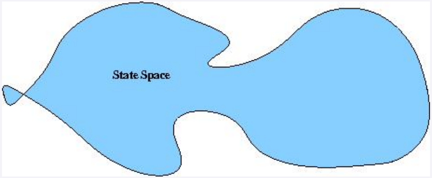
$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \xrightarrow{t_4} \left(m_3, \begin{pmatrix} 2.3 \\ \sharp \\ 0.0 \\ \sharp \end{pmatrix}\right) \xrightarrow{2.0} \left(m_4, \begin{pmatrix} 4.3 \\ \sharp \\ 2.0 \\ \sharp \end{pmatrix}\right)$$

Time Petri Nets and Turing Machines

Remark:

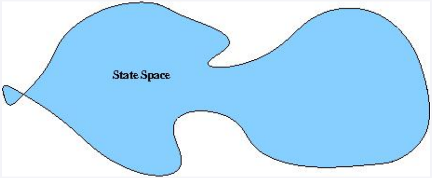
Time Petri Nets are Turing-complete.

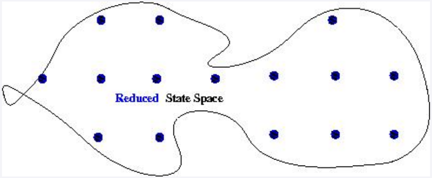
Some Problems: The State Space



The set of all reachable states is dense.

Discrete Reduction of the State Space

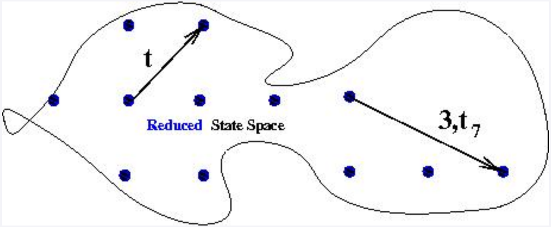




The set of all reachable states

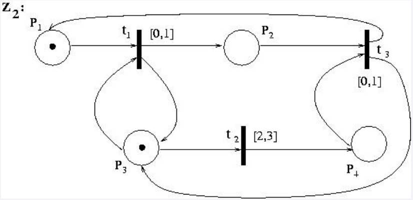
The set of all essential states

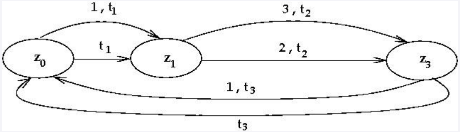
(Reduced) Reachability Graph



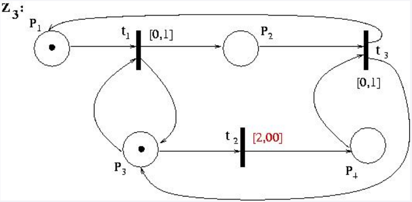
The reachability graph is a weighted directed graph, including the time explicit.

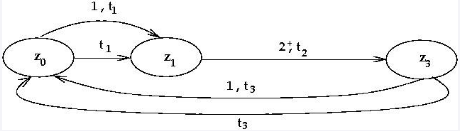
Example: A finite TPN and its reachability graph





Example: A non-finite TPN and its reachability graph





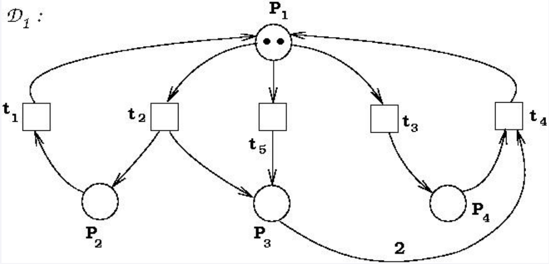
Time-dependent analysis

- qualitative analysis: using Model Checking (RG), ...
- quantitative analysis: using RG, linear programming (parametrization), ...

Timed Petri Net: An Informal Introduction

Statics:

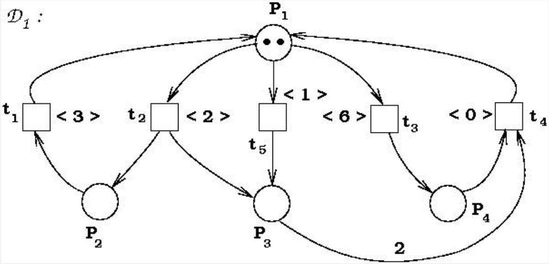
Petri Net



Timed Petri Net: An Informal Introduction

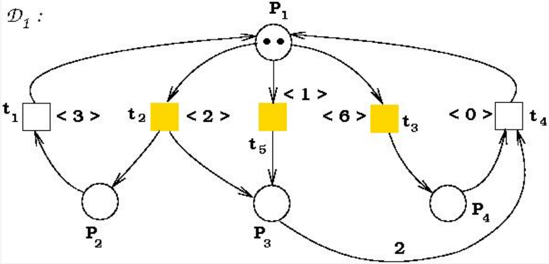
Statics:

Timed Petri Net



Timed Petri Net: An Informal Introduction

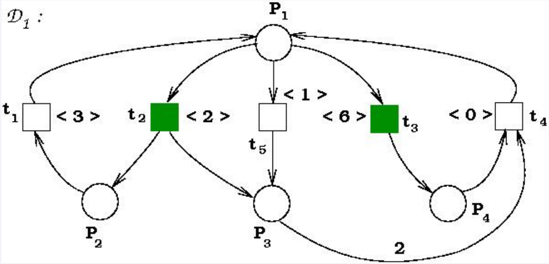
Dynamics:



firing mode: maximal step

Timed Petri Net: An Informal Introduction

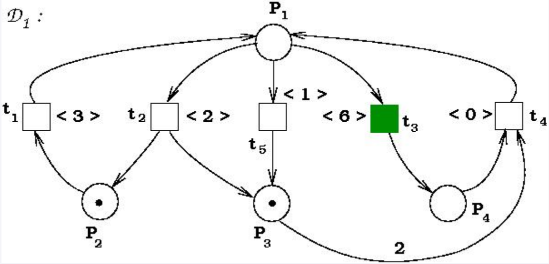
Dynamics:



firing mode: maximal step

Timed Petri Net: An Informal Introduction

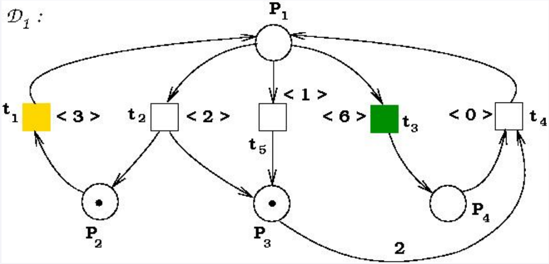
Dynamics:



firing mode: maximal step

Timed Petri Net: An Informal Introduction

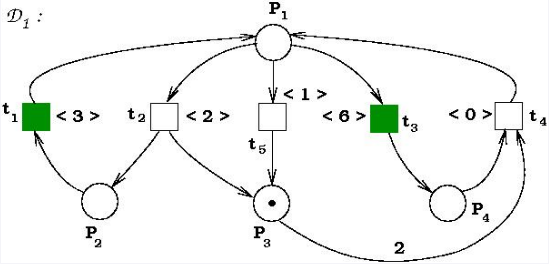
Dynamics:



firing mode: maximal step

Timed Petri Net: An Informal Introduction

Dynamics:



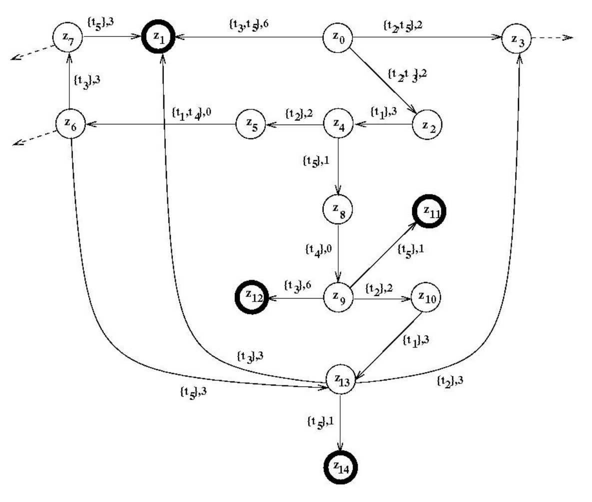
firing mode: maximal step

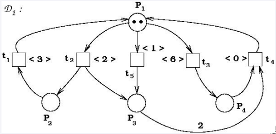
Timed Petri Nets and Turing Machines

Remark:

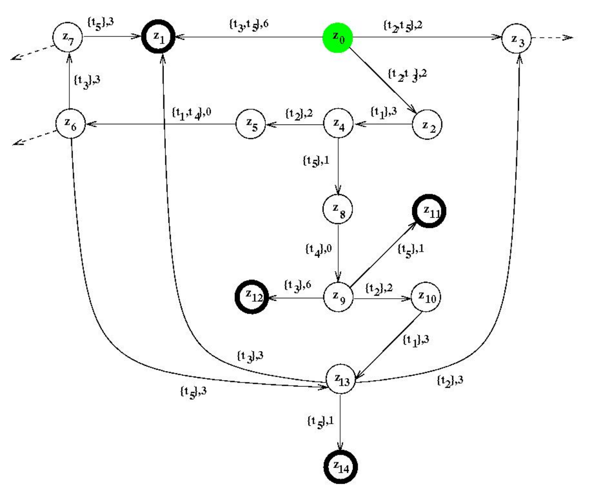
Timed Petri Nets are Turing-complete.

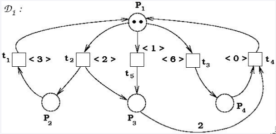
Reachability graph



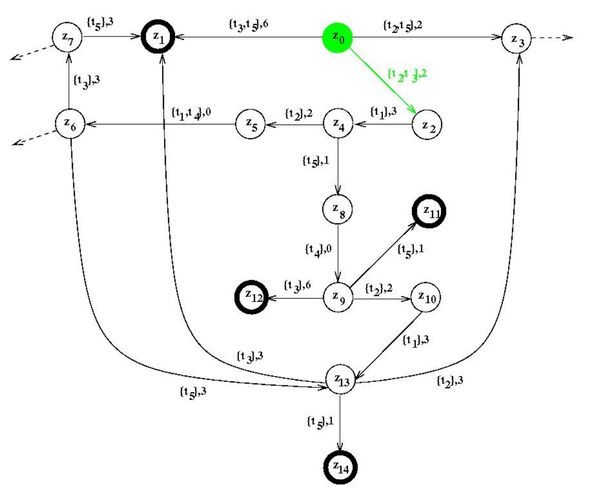


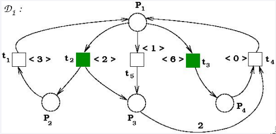
Reachability graph



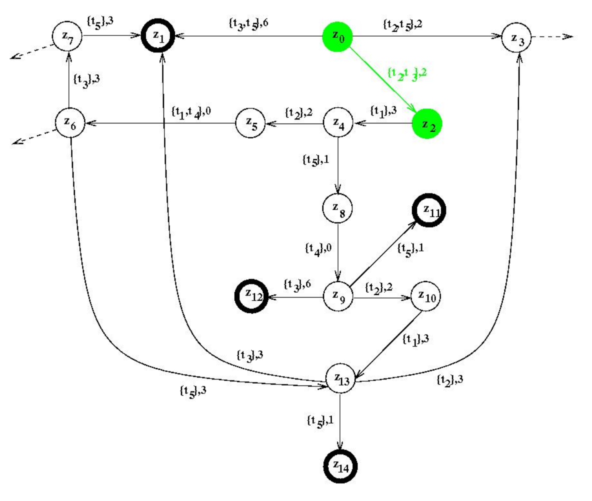


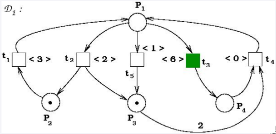
Reachability graph



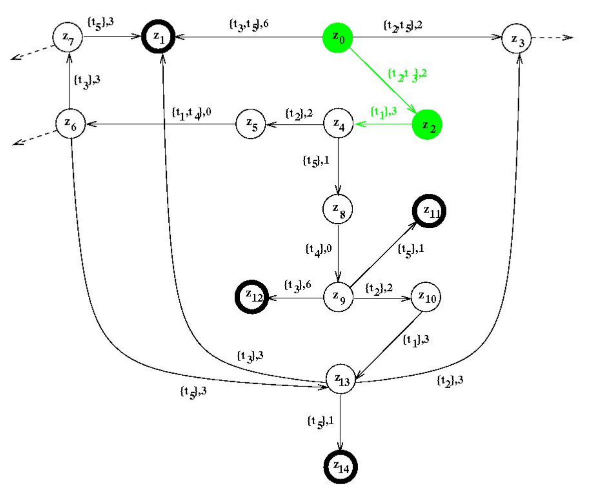


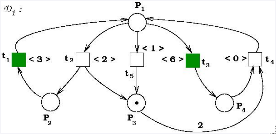
Reachability graph



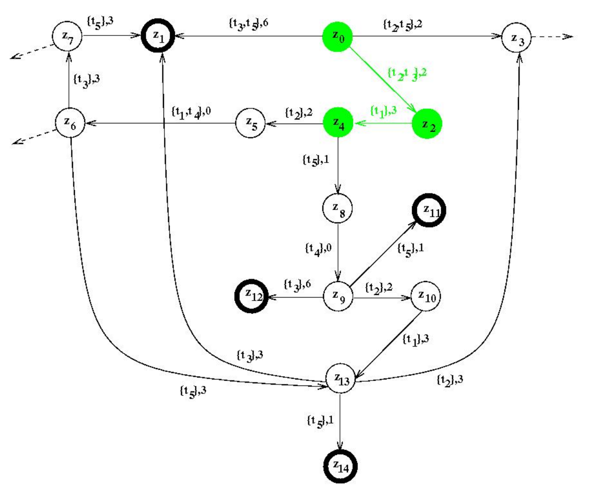


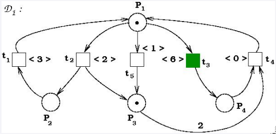
Reachability graph



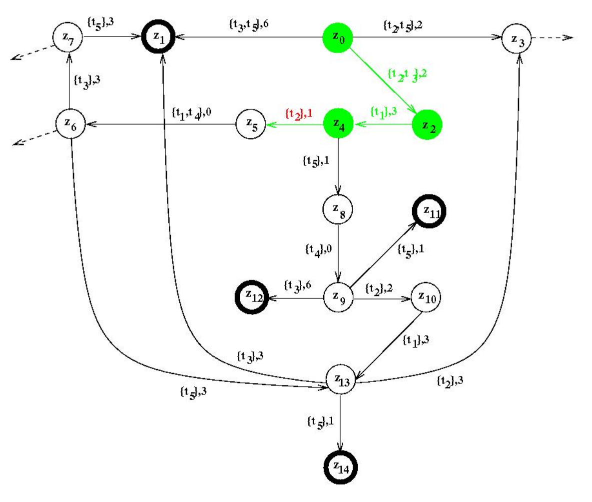


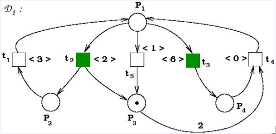
Reachability graph



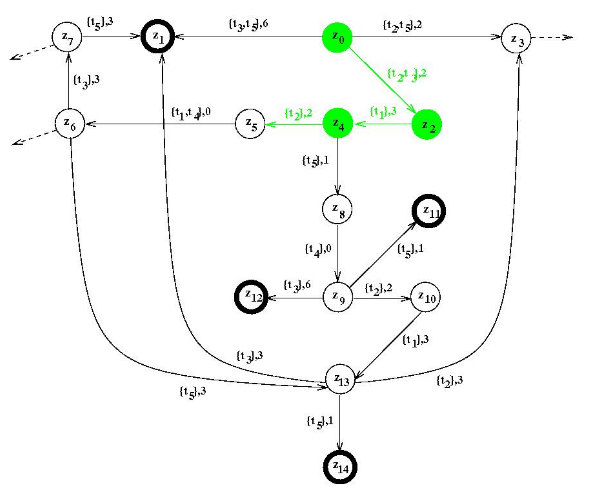


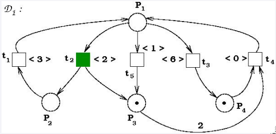
Reachability graph



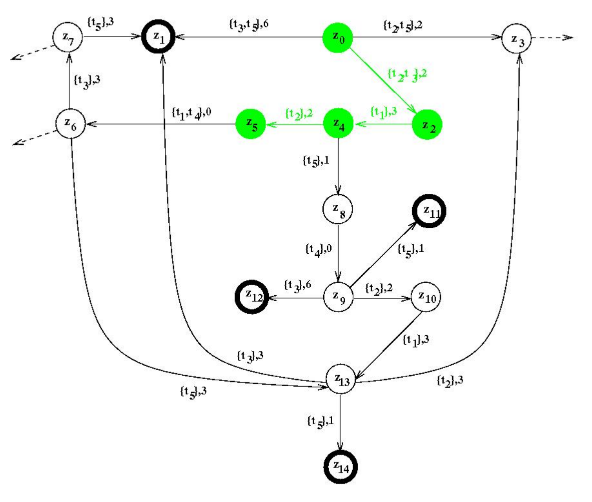


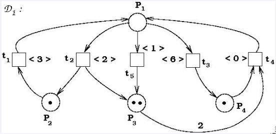
Reachability graph



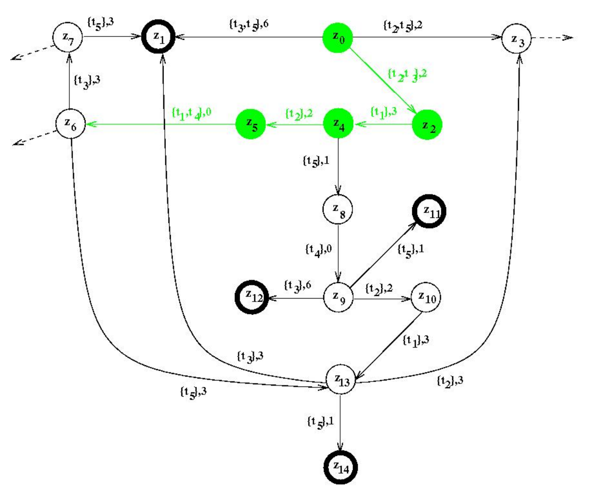


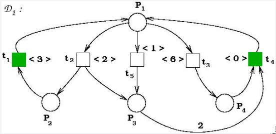
Reachability graph





Reachability graph





State Equation in classic PN

Let $\mathcal{N}$ be a classic PN with

- m_1 and m_2 two markings in $\mathcal{N}$,
- $\sigma = t_1 \ldots t_n$ a firing sequence, and
- $m_1 \xrightarrow{\sigma} m_2$.

Then it holds:

$$m_2 = m_1 + C \cdot \pi_\sigma, \text{ (state equation)}$$

where C is the incidence matrix of $\mathcal{N}$ and π_σ is the Parikh vector of σ.

State Equation in classic PN

Let $\mathcal{N}$ be a classic PN with

- m_1 and m_2 two markings in $\mathcal{N}$,
- $\sigma = t_1 \dots t_n$ a firing sequence, and
- $m_1 \xrightarrow{\sigma} m_2$.

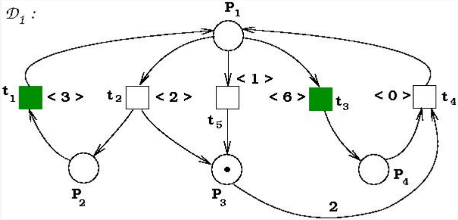
Then it holds:

$$m_2 = m_1 + C \cdot \pi_\sigma, \text{ (state equation)}$$

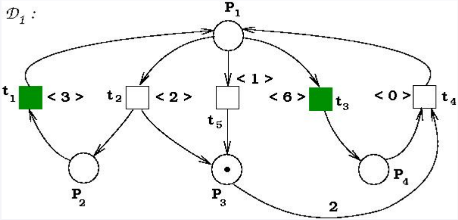
where C is the incidence matrix of $\mathcal{N}$ and π_σ is the Parikh vector of σ.

In each PN $\mathcal{N}$ with initial marking m_0 it holds:
If $m \neq m_0 + C \cdot \pi_\sigma$ for each π_σ then m is not reachable in $\mathcal{N}$.

Extended Form of a Place Marking



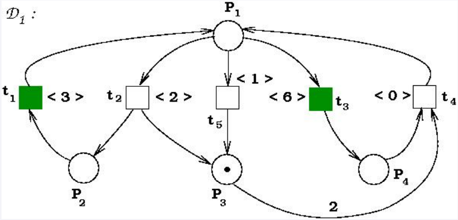
Extended Form of a Place Marking



$$m = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

extended form
of the p-markings m

Extended Form of a Place Marking



$$m = \begin{array}{c} \left(\begin{array}{ccccccc} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array} \right) \begin{array}{c} p_1 \\ p_2 \\ p_3 \\ p_4 \end{array} \\ \text{after} \\ \begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \text{time units} \end{array}$$

extended form of the p-markings m

Time Dependent State Equation

Theorem

Let $\mathcal{D}$ be a Timed Petri Net, $z^{(0)}$ be the initial state in extended form and

$$z^{(0)} \xrightarrow{\mathfrak{G}_1} \hat{z}^{(1)} \xrightarrow[1]{} \tilde{z}^{(1)} \xrightarrow{\mathfrak{G}_2} \hat{z}^{(2)} \xrightarrow[1]{} \ldots \xrightarrow{\mathfrak{G}_n} z^{(n)}$$

be a firing sequence ($\mathfrak{G}_i$ is a multiset for each i). Then, it holds:

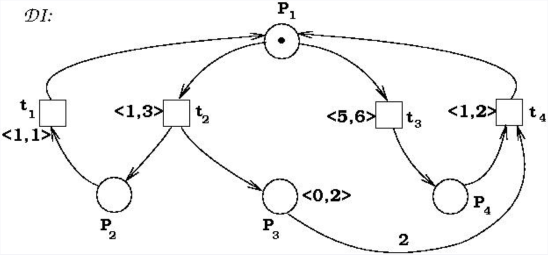
$$m^{(n)} = m^{(0)} \cdot R^{n-1} + C \cdot \Psi_\sigma. \qquad \text{State equation}$$

$$z^{(0)} \xrightarrow{\mathfrak{G}_1} \hat{z}^{(1)} \xrightarrow[1]{} \tilde{z}^{(1)} \xrightarrow{\mathfrak{G}_2} \hat{z}^{(2)} \xrightarrow[1]{} \ldots \xrightarrow{\mathfrak{G}_n} z^{(n)}$$

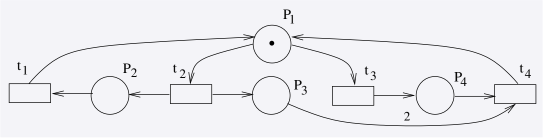
$$m^{(n)} = m^{(0)} \cdot R^{n-1} + C \cdot \Psi_\sigma. \qquad \text{State equation}$$

- $m^{(n)}$ and $m^{(0)}$ are place markings in extended form
- R is the progress matrix for $\mathcal{D}$.
- C is the incidence matrix of $\mathcal{D}$ in extended form
- Ψ_σ is the Parikh matix of the sequence $\sigma = \mathfrak{G}_1\ \mathfrak{G}_2\ \ldots \mathfrak{G}_n$ of multisets of transitions.

Timed Petri Nets with Variable Durations: An Informal Introduction

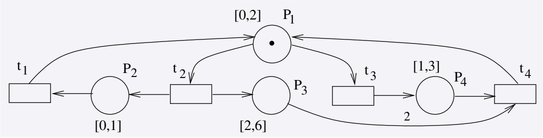


Petri Nets with Time Windows (tw-PN): An Informal Introduction



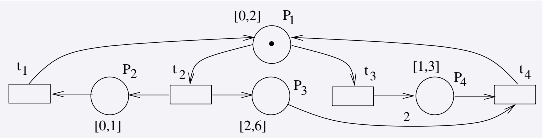
A **Petri Net with Time Windows** $\mathcal{P} = (\mathcal{N}, \mathcal{I})$
is a Petri net $\mathcal{N}$

Petri Nets with Time Windows (tw-PN): An Informal Introduction



A **Petri Net with Time Windows** $\mathcal{P} = (\mathcal{N}, \mathcal{I})$
is a Petri net $\mathcal{N}$
with time intervals (windows) attached to the places.

Initial Time Marking (Example)



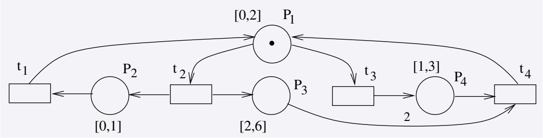
The initial time marking is given by

$$M_0 = (\overbrace{0}^{M(p_1)}; \overbrace{\varepsilon}^{M(p_2)}; \overbrace{\varepsilon}^{M(p_3)}; \overbrace{\varepsilon}^{M(p_4)})$$

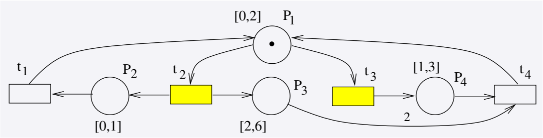
the initial (timeless) marking by

$$m_{M_0} = (1; 0; 0; 0) = m_0$$

Example: Firing a transition t



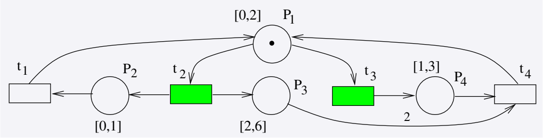
Example: Firing a transition t



"enough" tokens on pre-places of t

$\Rightarrow$ transition t enabled

Example: Firing a transition t



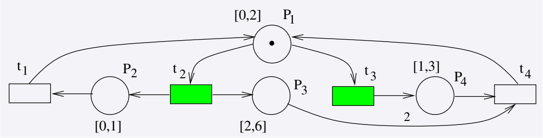
"enough" tokens on pre-places of t

$\Rightarrow$ transition t enabled

all needed tokens "old enough"

$\Rightarrow$ transition t ready to fire

Example: Firing a transition *t*



"enough" tokens on pre-places of t

$\Rightarrow$ transition t enabled

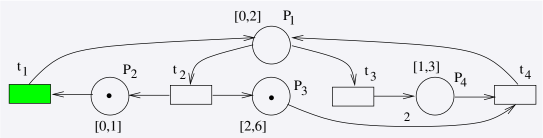
all needed tokens "old enough"

$\Rightarrow$ transition t ready to fire

$M_0 = (0, \varepsilon, \varepsilon, \varepsilon)$

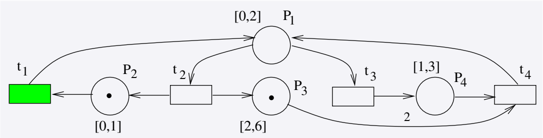
$\Rightarrow$ t_2 and t_3: enabled and ready to fire

Example: Firing a transition t



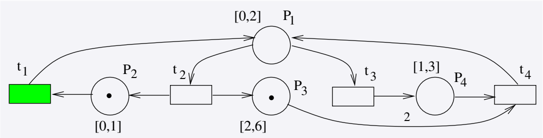
$$M_0 \xrightarrow{t_2} M_1 = (\varepsilon, 0, 0, \varepsilon)$$

Example: Firing a transition t



$$M_0 \xrightarrow{t_2} M_1 = (\varepsilon, 0, 0, \varepsilon)$$
$$M_1 \xrightarrow{1} M_2 = (\varepsilon, 1, 1, \varepsilon)$$

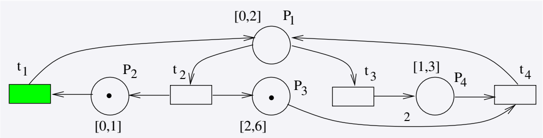
Example: Firing a transition *t*



$$M_0 \xrightarrow{t_2} M_1 = (\varepsilon, 0, 0, \varepsilon)$$
$$M_1 \xrightarrow{1} M_2 = (\varepsilon, 1, 1, \varepsilon)$$

A transition is not forced to fire!

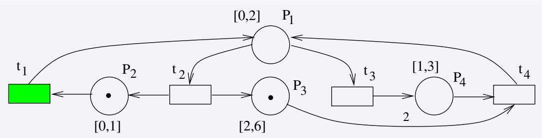
Example: Firing a transition t



$M_0 \xrightarrow{t_2} M_1 = (\varepsilon, 0, 0, \varepsilon)$
$M_1 \xrightarrow{1} M_2 = (\varepsilon, 1, 1, \varepsilon)$

A transition is not forced to fire!
The age is reset when the retention time is greater than upper time bound.

Example: Firing a transition t



$$M_0 \xrightarrow{t_2} M_1 = (\varepsilon, 0, 0, \varepsilon)$$
$$M_1 \xrightarrow{1} M_2 = (\varepsilon, 1, 1, \varepsilon)$$
$$M_2 \xrightarrow{0.5} M_3 = (\varepsilon, 0.5, 1.5, \varepsilon)$$

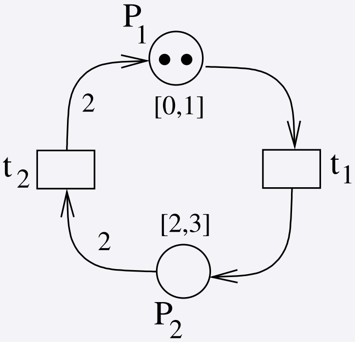
A transition is not forced to fire!
The age is reset when the retention time is greater than upper time bound.

tw-Petri Nets and Turing Machines

Remark:

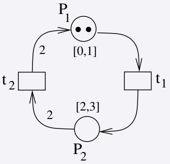
The tw-PNs are not Turing-complete.

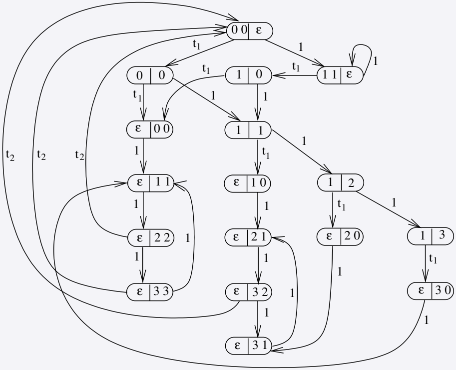
Reachability Graph: Natural Numbers vs. Real Numbers



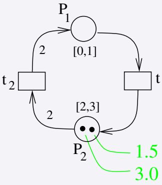
Reachability Graph: Natural Numbers vs. Real Numbers

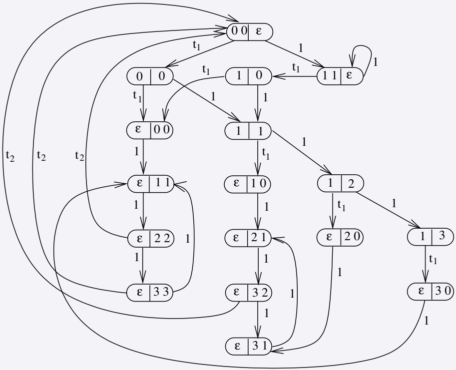
There is no “leaf” in the integer reachability graph!





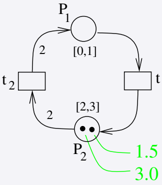
Reachability Graph: Natural Numbers vs. Real Numbers

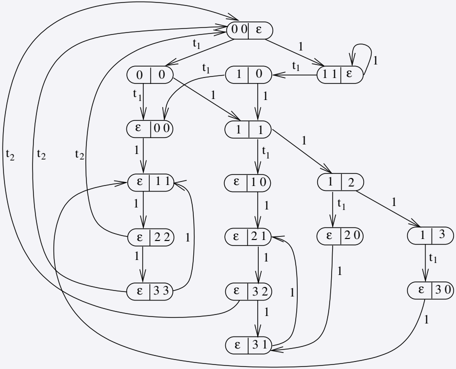




Consider $\sigma(\tau) = t_1\ 1.5\ t_1$

Reachability Graph: Natural Numbers vs. Real Numbers





Consider $\sigma(\tau) = t_1\ 1.5\ t_1\ 0.5\ 1.0\ 0.5\ 1.0$

$\Rightarrow t_2$ is in $M = (\varepsilon, 3.0\ 1.5)$ in a t-DL

Properties

Property "Reachability"

A marking M is reachable in a tw-PN $\mathcal{P}$ iff m_M is reachable in $S(\mathcal{P})$.

Properties

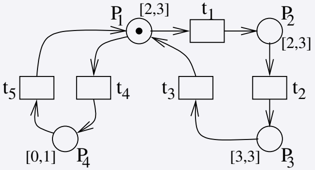
Property “Reachability”

A marking M is reachable in a tw-PN $\mathcal{P}$ iff m_M is reachable in $S(\mathcal{P})$.

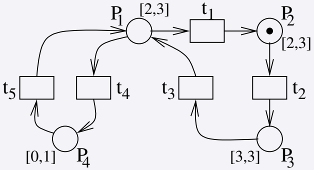
Property “Liveness”

There is not a correlation between the liveness behaviors of a tw-PN and its skeleton.

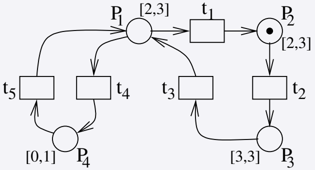
Time Gaps

	
$\sigma(\tau_\alpha) = 3$	

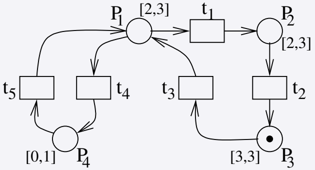
Time Gaps

<table>
<tr><td>
</td><td></td></tr>
<tr><td>$\sigma(\tau_\alpha) = 3\ t_1$</td><td></td></tr>
</table>

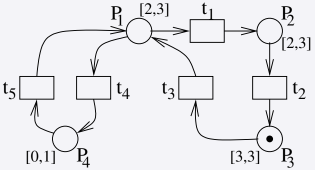
Time Gaps

	
$\sigma(\tau_\alpha) = 3\ t_1\ 3$	

Time Gaps

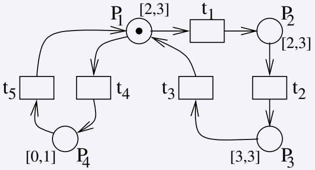
	
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2$	

Time Gaps



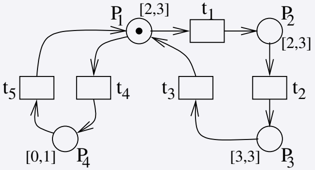
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3$

Time Gaps



$$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$$

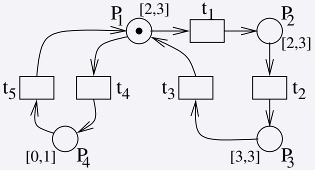
Time Gaps

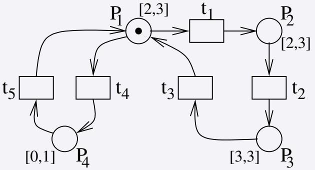


$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$

$\Rightarrow \alpha = 9$

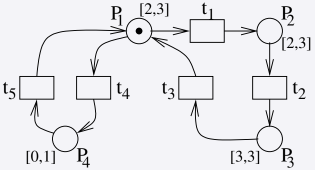
Time Gaps

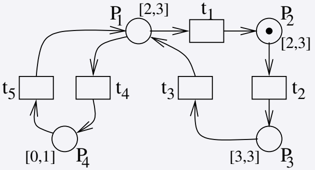




$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5$

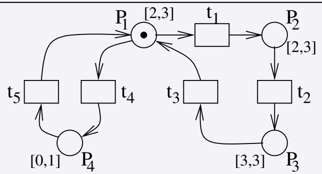
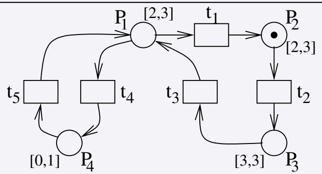
Time Gaps



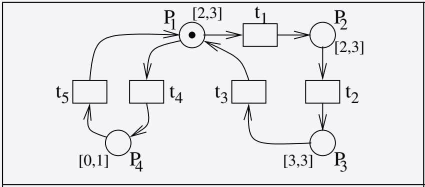


$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5\ t_1$

Time Gaps

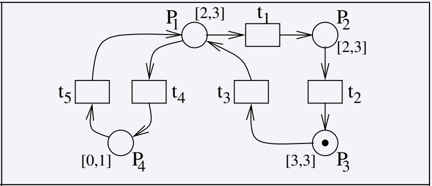
	
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5\ t_1\ 2$

Time Gaps



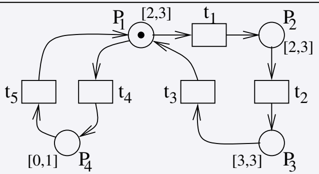
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$

$\Rightarrow \alpha = 9$



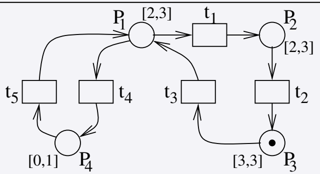
$\sigma(\tau_\beta) = 5\ t_1\ 2\ t_2$

Time Gaps



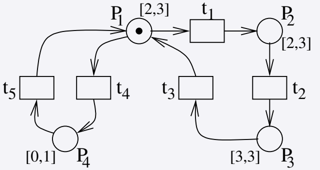
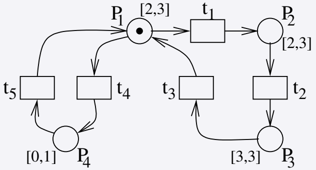
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$

$\Rightarrow \alpha = 9$

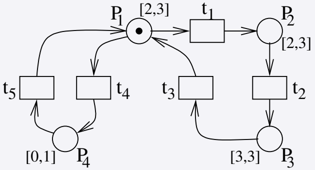
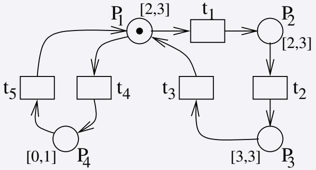


$\sigma(\tau_\beta) = 5\ t_1\ 2\ t_2\ 3$

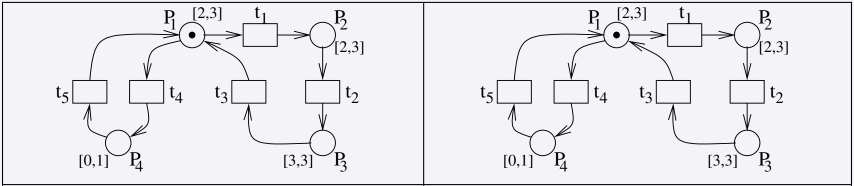
Time Gaps

	
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5\ t_1\ 2\ t_2\ 3\ t_3$

Time Gaps

	
$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5\ t_1\ 2\ t_2\ 3\ t_3$ $\Rightarrow \beta = 10$

Time Gaps



$\sigma(\tau_\alpha) = 3\ t_1\ 3\ t_2\ 3\ t_3$ $\Rightarrow \alpha = 9$	$\sigma(\tau_\beta) = 5\ t_1\ 2\ t_2\ 3\ t_3$ $\Rightarrow \beta = 10$

$\cancel{\gamma = 9.5?}$

- Given: Time dependent Petri Net
- Aim: Analysis of the time dependent Petri Net
- Problem: Infinite (dense) state space, Turing-completeness
- Solution:
    - Parametrisation and discretisation of the state space.
    - Definition of an reachability graph.
    - Structurally restricted classes of time dependent Petri Nets.
    - Time dependent state equation.
- Remark: The time is not the reason for Turing-completeness of a time-dependent Petri net.

More about time and Petri nets in