A Memo on Computability in Time Petri Net

Louchka Popova-Zeugmann

Humboldt-Universität zu Berlin
Institut of Computer Science
Unter den Linden 6, 10099 Berlin, Germany

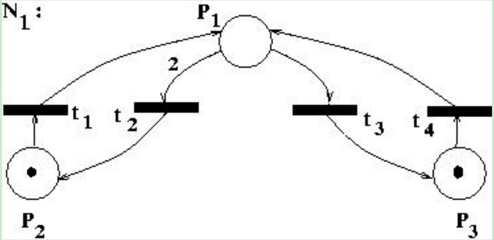
CS&P 2005
Ruciane-Nida, Poland
September 28-30 2005

Berlin - Brandenburger Tor

Outline

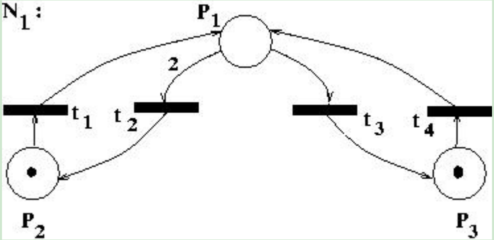
Petri Net

Example



Petri Net

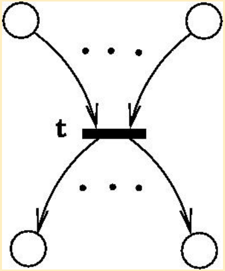
Example



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

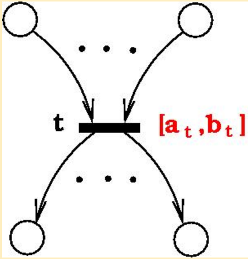
Time Petri Net

Definition (informal)



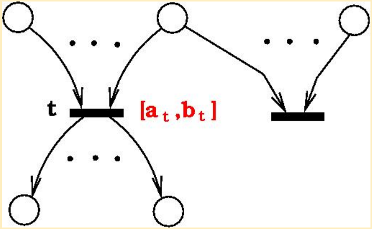
Time Petri Net

Definition (informal)



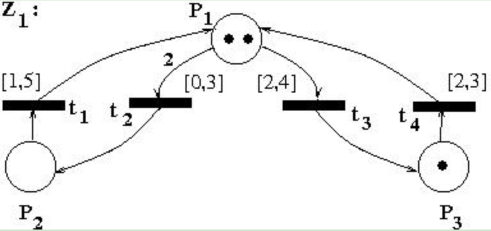
Time Petri Net

Definition (informal)



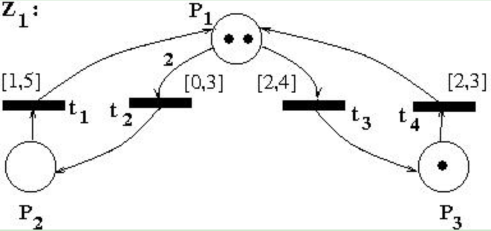
Time Petri Net

Example



Time Petri Net

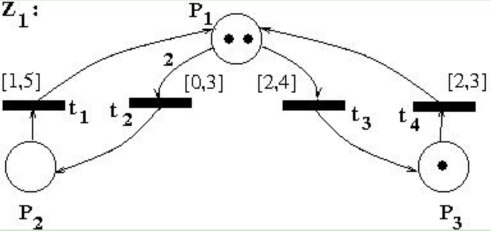
Example



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

Time Petri Net

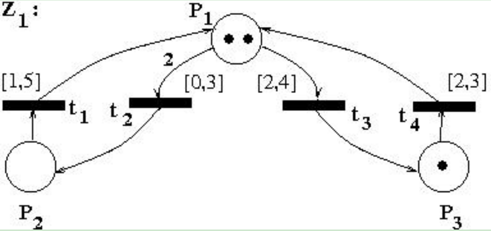
Example



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

Time Petri Net

Example



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

state

Definition (state)

$z = (m, h)$ is called a **state** in a TPN Z iff:

state

Definition (state)

$z = (m, h)$ is called a **state** in a TPN Z iff:

- m is a p-marking in Z .

state

Definition (state)

$z = (m, h)$ is called a **state** in a TPN Z iff:

- m is a p-marking in Z .
- h is a t-marking in Z.

Definition (state changing)

Definition (state changing)

Let Z be a TPN, and $z = (m, h)$, $z' = (m', h')$ be two states.

Definition (state changing)

Let Z be a TPN, and $z = (m, h)$, $z' = (m', h')$ be two states. Then

$$z = (m, h) \text{ can change into } z' = (m', h')$$

Definition (state changing)

Let Z be a TPN, and $z = (m, h)$, $z' = (m', h')$ be two states. Then

$$z = (m, h) \text{ can change into } z' = (m', h') \text{ by}$$

firing
a transition /

Definition (state changing)

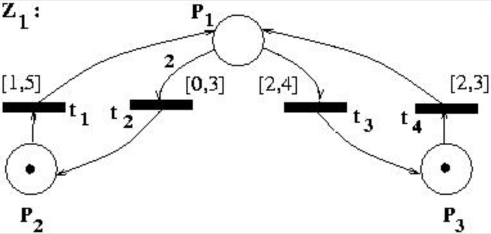
Let Z be a TPN, and $z = (m, h)$, $z' = (m', h')$ be two states. Then

$$z = (m, h) \text{ can change into } z' = (m', h') \text{ by}$$

firing a transition / \ **time elapsing**

Time Net

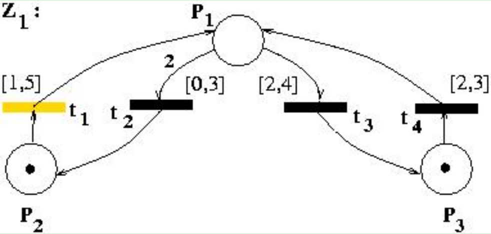
Example



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

Time Net

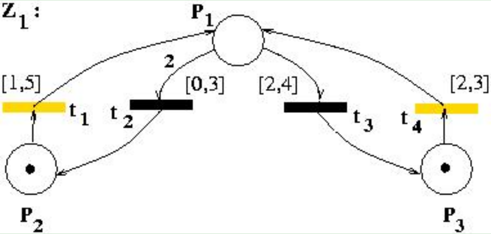
Example



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

Time Net

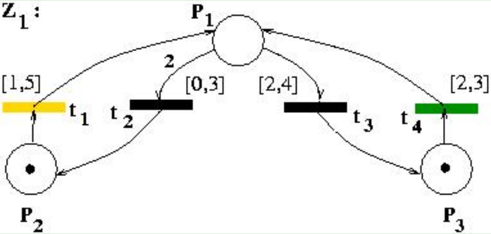
Example



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

Time Net

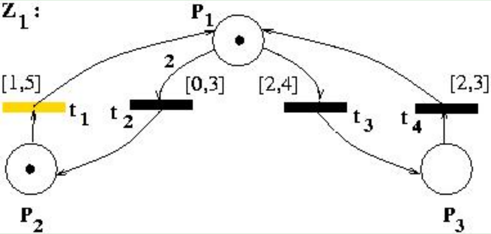
Example



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

Time Net

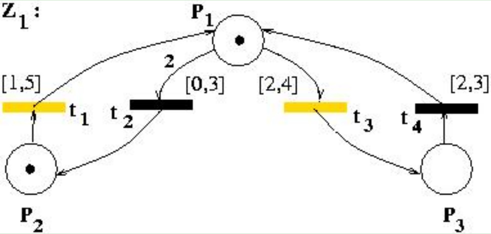
Example



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

Time Net

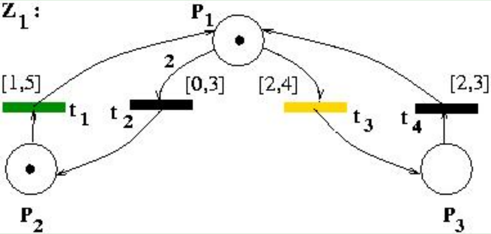
Example



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

Time Net

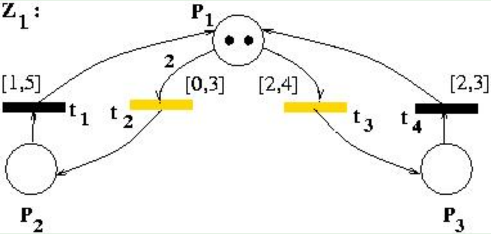
Example



$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \xrightarrow{t_4} \xrightarrow{2.0} \left(m_4, \begin{pmatrix} 4.3 \\ \sharp \\ 2.0 \\ \sharp \end{pmatrix}\right) \xrightarrow{t_1}$$

Time Net

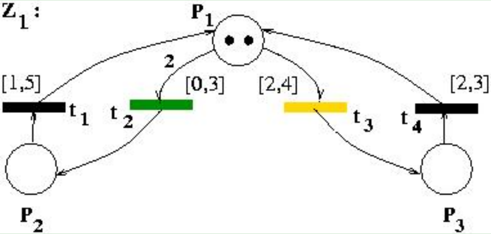
Example



$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \xrightarrow{t_4} \xrightarrow{2.0} (m_4, \begin{pmatrix} 4.3 \\ \sharp \\ 2.0 \\ \sharp \end{pmatrix}) \xrightarrow{t_1} (m_5, \begin{pmatrix} \sharp \\ 0.0 \\ 2.0 \\ \sharp \end{pmatrix})$$

Time Net

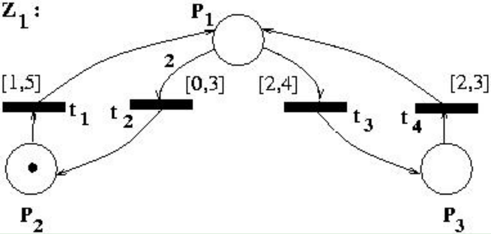
Example



$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \xrightarrow{t_4} \xrightarrow{2.0} \xrightarrow{t_1} \left(m_5, \begin{pmatrix} \sharp \\ 0.0 \\ 2.0 \\ \sharp \end{pmatrix}\right) \xrightarrow{t_2}$$

Time Net

Example



$$z_0 \xrightarrow{1.3} \xrightarrow{1.0} \xrightarrow{t_4} \xrightarrow{2.0} \xrightarrow{t_1} \left(m_5, \begin{pmatrix} \sharp \\ 0.0 \\ 2.0 \\ \sharp \end{pmatrix}\right) \xrightarrow{t_2} \left(m_6, \begin{pmatrix} 0.0 \\ \sharp \\ \sharp \\ \sharp \end{pmatrix}\right)$$

Transition sequences, Runs

Definition

- **transition sequence:** $\sigma = (t_1, \cdots, t_n)$

Transition sequences, Runs

Definition

- **transition sequence:** $\sigma = (t_1, \cdots, t_n)$
- **run:** $\sigma(\tau) = (\tau_0, t_1, \tau_1, \cdots, \tau_{n-1}, t_n, \tau_n)$

Transition sequences, Runs

Definition

- **transition sequence:** $\sigma = (t_1, \cdots, t_n)$
- **run:** $\sigma(\tau) = (\tau_0, t_1, \tau_1, \cdots, \tau_{n-1}, t_n, \tau_n)$
- **feasible run:** $z_0 \xrightarrow{\tau_0} z_0^* \xrightarrow{t_1} z_1 \xrightarrow{\tau_1} z_1^* \cdots \xrightarrow{t_n} z_n \xrightarrow{\tau_n} z_n^*$

Transition sequences, Runs

Definition

- **transition sequence:** $\sigma = (t_1, \cdots, t_n)$
- **run:** $\sigma(\tau) = (\tau_0, t_1, \tau_1, \cdots, \tau_{n-1}, t_n, \tau_n)$
- **feasible run:** $z_0 \xrightarrow{\tau_0} z_0^* \xrightarrow{t_1} z_1 \xrightarrow{\tau_1} z_1^* \cdots \xrightarrow{t_n} z_n \xrightarrow{\tau_n} z_n^*$
- **feasible transition sequence :** σ is feasible if there ex. a feasible run $\sigma(\tau)$

Reachable state, Reachable marking, State space

Definition

- z is **reachable state** in Z if there ex. a feasible run $\sigma(\tau)$ and $z_0 \xrightarrow{\sigma(\tau)} z$

Reachable state, Reachable marking, State space

Definition

- z is **reachable state** in Z if there ex. a feasible run $\sigma(\tau)$ and $z_0 \xrightarrow{\sigma(\tau)} z$
- The set of all reachable states in Z is the **state space** of Z (denoted: $StSp(Z)$).

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.
$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.
$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

- $m_0 \xrightarrow{\sigma} m_\sigma$

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.
$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

- $m_0 \xrightarrow{\sigma} m_\sigma$
- $\Sigma_\sigma(t)$ is a sum of variables,
  Σ_σ is a parametrical $t-$marking

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.

$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

- $m_0 \xrightarrow{\sigma} m_\sigma$
- $\Sigma_\sigma(t)$ is a sum of variables,
  Σ_σ is a parametrical $t-$marking
- B_σ is a set of conditions (a system of inequalities)

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.
$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

- $m_0 \xrightarrow{\sigma} m_\sigma$
- $\Sigma_\sigma(t)$ is a sum of variables,
  Σ_σ is a parametrical $t-$marking
- B_σ is a set of conditions (a system of inequalities)

Obviously

- $z_0 \xrightarrow{\sigma} (m_\sigma, \Sigma_\sigma) =: z_\sigma$,

Parametric Description of the State Space

Let $Z = [P, T, F, V, m_0, I]$ be a TPN and $\sigma = (t_1, \cdots, t_n)$ be a transition sequence in Z.

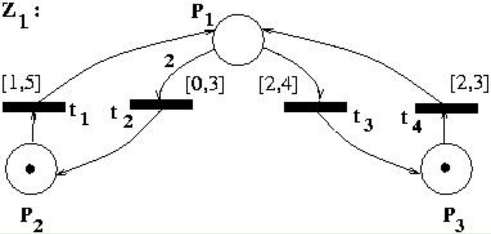
$\delta(\sigma) = [m_\sigma, \Sigma_\sigma, B_\sigma]$ is the parametric description of σ, if

- $m_0 \xrightarrow{\sigma} m_\sigma$
- $\Sigma_\sigma(t)$ is a sum of variables,
  Σ_σ is a parametrical $t-$marking
- B_σ is a set of conditions (a system of inequalities)

Obviously

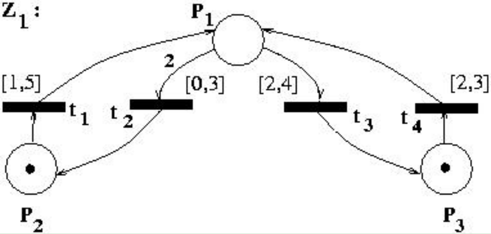
- $z_0 \xrightarrow{\sigma} (m_\sigma, \Sigma_\sigma) =: z_\sigma$,
- $StSp(Z) = \bigcup\limits_{\sigma} z_\sigma$.

Example



$$\sigma = (t_4, t_3)$$

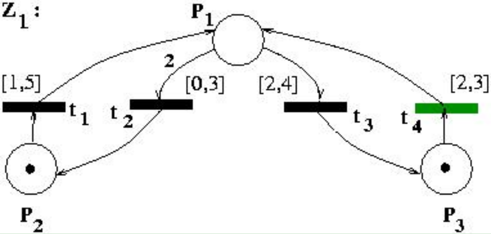
Example



$$\sigma = (e) \quad \Longrightarrow \quad \delta(\sigma) =$$

$$\left\{\left(\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ \sharp \\ \sharp \\ x_1 \end{pmatrix}\right) \;\middle|\; 0 \le x_1 \le 3 \right\}.$$

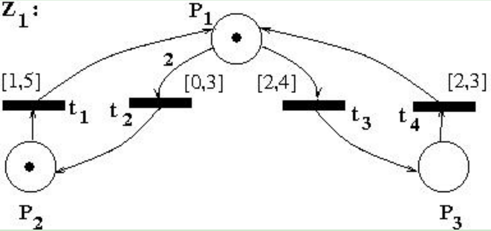
Example



$$\sigma = (e) \quad \Longrightarrow \quad \delta(\sigma) =$$

$$\{ (\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ \sharp \\ \sharp \\ x_1 \end{pmatrix}) \mid \ 0 \leq x_1 \leq 3 \ \}.$$

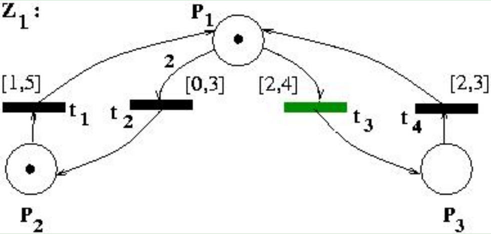
Example



$$\sigma = (t_4) \quad \Longrightarrow \quad \delta(\sigma) =$$

$$\{(\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} x_1 + x_2 \\ \sharp \\ x_2 \\ \sharp \end{pmatrix}) \mid \begin{array}{ll} 2 \leq x_1 \leq 3, & x_1 + x_2 \leq 5 \\ 0 \leq x_2 \leq 4, & \end{array} \}.$$

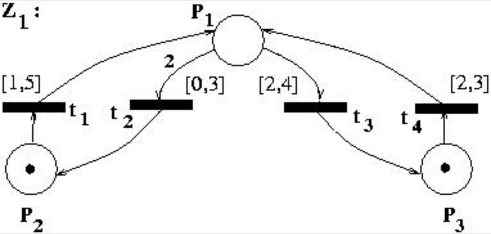
Example



$$\sigma = (t_4) \qquad \Longrightarrow \qquad \delta(\sigma) =$$

$$\{(\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} x_1 + x_2 \\ \sharp \\ x_2 \\ \sharp \end{pmatrix}) \mid \begin{array}{ll} 2 \leq x_1 \leq 3, & x_1 + x_2 \leq 5 \\ 0 \leq x_2 \leq 4, & \end{array} \}.$$

Example

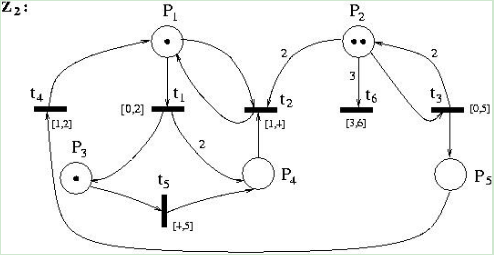


$$\sigma = (t_4, t_3) \quad \Longrightarrow \quad \delta(\sigma) =$$

$$\left\{\left(\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 + x_2 + x_3 \\ \sharp \\ \sharp \\ x_3 \end{pmatrix} \right) \;\middle|\; \begin{array}{ll} 2 \le x_1 \le 3, & x_1 + x_2 \le 5 \\ 2 \le x_2 \le 4, & x_1 + x_2 + x_3 \le 5 \\ 0 \le x_3 \le 3 & \end{array} \right\}.$$

State Space Reduction

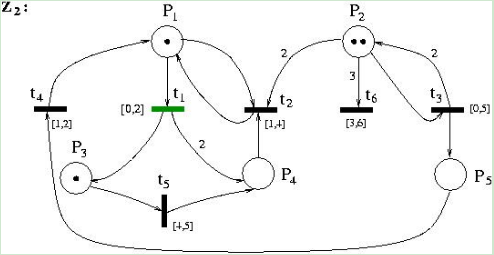
Example



$$\sigma = (t_1 \;\; t_3 \;\; t_4 \;\; t_2 \;\; t_3)$$

State Space Reduction

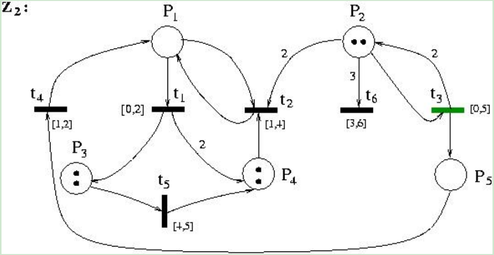
Example



$$\sigma = (t_1 \ t_3 \ t_4 \ t_2 \ t_3)$$

State Space Reduction

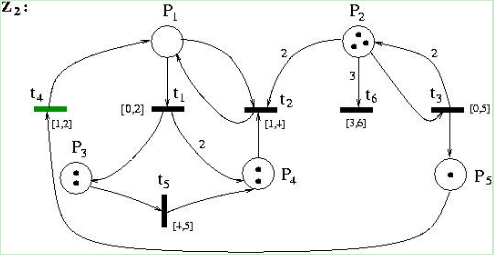
Example



$$\sigma = (t_1\ t_3\ t_4\ t_2\ t_3)$$

State Space Reduction

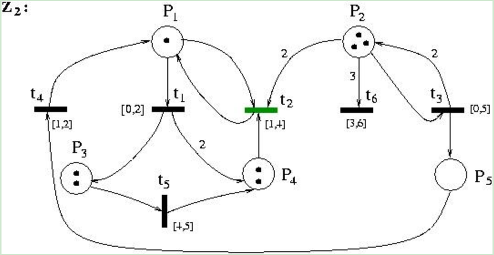
Example



$$\sigma = (t_1 \;\; t_3 \;\; t_4 \;\; t_2 \;\; t_3)$$

State Space Reduction

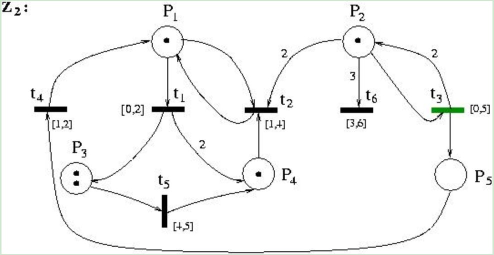
Example



$$\sigma = (t_1 \;\; t_3 \;\; t_4 \;\; \color{green}{t_2} \;\; t_3)$$

State Space Reduction

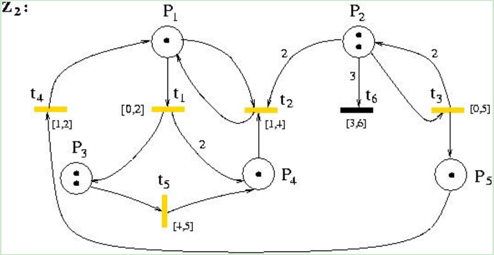
Example



$$\sigma = (t_1\ \ t_3\ \ t_4\ \ t_2\ \ t_3)$$

State Space Reduction

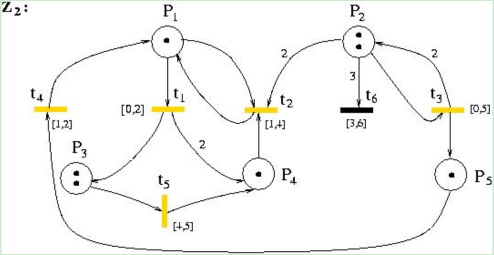
Example



$$\sigma = (t_1 \;\; t_3 \;\; t_4 \;\; t_2 \;\; t_3)$$

State Space Reduction

Example

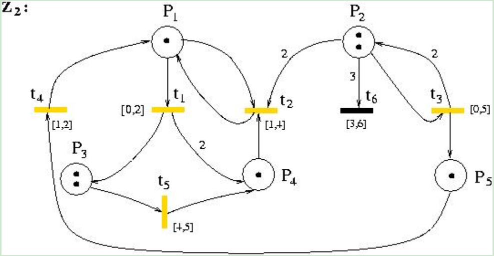


$$\sigma = (t_1 \;\; t_3 \;\; t_4 \;\; t_2 \;\; t_3)$$

$$\sigma(\tau) := z_0 \xrightarrow{0.7} \xrightarrow{t_1} \xrightarrow{0.0} \xrightarrow{t_3} \xrightarrow{0.4} \xrightarrow{t_4} \xrightarrow{1.2} \xrightarrow{t_2} \xrightarrow{0.5} \xrightarrow{t_3} \xrightarrow{1.4} z$$

State Space Reduction

Example

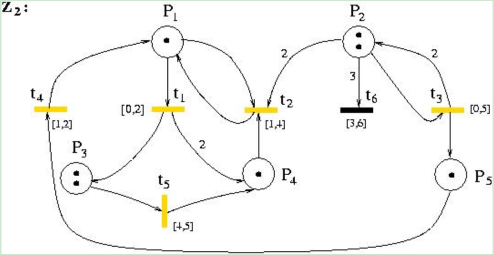


$$\sigma = (t_1 t_3 t_4 t_2 t_3)$$

$$\sigma(\tau) := z_0 \xrightarrow{\mathbf{0.7}} \xrightarrow{t_1} \xrightarrow{\mathbf{0.0}} \xrightarrow{t_3} \xrightarrow{\mathbf{0.4}} \xrightarrow{t_4} \xrightarrow{\mathbf{1.2}} \xrightarrow{t_2} \xrightarrow{\mathbf{0.5}} \xrightarrow{t_3} \xrightarrow{\mathbf{1.4}} z$$

State Space Reduction

Example

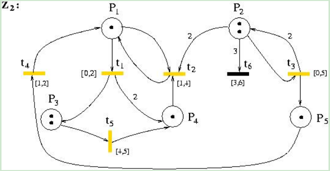


$$\sigma = (t_1 t_3 t_4 t_2 t_3)$$

$$m_\sigma = (1, 2, 2, 1, 1)$$

State Space Reduction

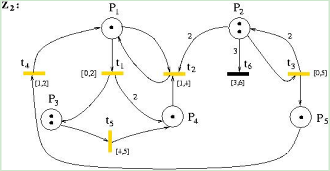
Example (continuation)



$$\Sigma_\sigma = \begin{pmatrix} x_4 + x_5 \\ x_5 \\ x_5 \\ x_5 \\ x_0 + x_1 + x_2 + x_3 + x_4 + x_5 \\ \sharp \end{pmatrix} \quad \text{and}$$

State Space Reduction

Example (continuation)



$$B_\sigma = \left\{ \begin{array}{lll} 0 \le x_0, & x_0 \le 2, & x_0 + x_1 + x_2 \le 5 \\ 0 \le x_1, & x_2 \le 2, & x_2 + x_3 \le 5 \\ 1 \le x_2, & x_3 \le 2, & x_0 + x_1 + x_2 + x_3 \le 5 \\ 1 \le x_3, & x_4 \le 2, & x_0 + x_1 + x_2 + x_3 + x_4 \le 5 \\ 0 \le x_4, & x_5 \le 2, & x_0 + x_1 + x_2 + x_3 + x_4 + x_5 \le 5 \\ 0 \le x_5, & x_0 + x_1 \le 5 & x_4 + x_5 \le 2 \end{array} \right\}.$$

State Space Reduction

Example (continuation)

The run $\sigma(\tau)$ with
$\sigma(\tau) =$

$$z_0 \xrightarrow{\mathbf{0.7}} \xrightarrow{t_1} \xrightarrow{\mathbf{0.0}} \xrightarrow{t_3} \xrightarrow{\mathbf{0.4}} \xrightarrow{t_4} \xrightarrow{\mathbf{1.2}} \xrightarrow{t_2} \xrightarrow{\mathbf{0.5}} \xrightarrow{t_3} \xrightarrow{\mathbf{1.4}} \left(m_\sigma, \begin{pmatrix} 1.9 \\ 1.4 \\ 1.4 \\ 1.4 \\ 4.2 \\ \sharp \end{pmatrix}\right)$$

is feasible.

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} \mathbf{1.9} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{4.2} \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z}$$

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 4.0 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(?)} \lfloor z \rfloor} \qquad \underbrace{\left(m_\sigma, \begin{pmatrix} 1.9 \\ 1.4 \\ 1.4 \\ 1.4 \\ 4.2 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z}$$

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 4.0 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(?)} \lfloor z \rfloor} \qquad \underbrace{\left(m_\sigma, \begin{pmatrix} 1.9 \\ 1.4 \\ 1.4 \\ 1.4 \\ 4.2 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z} \qquad \underbrace{\left(m_\sigma, \begin{pmatrix} 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \\ 5.0 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(?)} \lceil z \rceil}$$

State Space Reduction

Example (continuation)

The runs

$$\sigma(\tau_1^*) := z_0 \xrightarrow{\mathbf{1}} \xrightarrow{t_1} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{1}} \xrightarrow{t_4} \xrightarrow{\mathbf{1}} \xrightarrow{t_2} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{1}} \lfloor z \rfloor$$

and

$$\sigma(\tau_2^*) := z_0 \xrightarrow{\mathbf{1}} \xrightarrow{t_1} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{0}} \xrightarrow{t_4} \xrightarrow{\mathbf{2}} \xrightarrow{t_2} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{2}} \lceil z \rceil$$

are feasible in Z, too.

State Space Reduction

Example (continuation)

The runs

$$\sigma(\tau_1^*) := z_0 \xrightarrow{\mathbf{1}} \xrightarrow{t_1} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{1}} \xrightarrow{t_4} \xrightarrow{\mathbf{1}} \xrightarrow{t_2} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{1}} \lfloor z \rfloor$$

$$\sigma(\tau) = z_0 \xrightarrow{\mathbf{0.7}} \xrightarrow{t_1} \xrightarrow{\mathbf{0.0}} \xrightarrow{t_3} \xrightarrow{\mathbf{0.4}} \xrightarrow{t_4} \xrightarrow{\mathbf{1.2}} \xrightarrow{t_2} \xrightarrow{\mathbf{0.5}} \xrightarrow{t_3} \xrightarrow{\mathbf{1.4}} z$$

$$\sigma(\tau_2^*) := z_0 \xrightarrow{\mathbf{1}} \xrightarrow{t_1} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{0}} \xrightarrow{t_4} \xrightarrow{\mathbf{2}} \xrightarrow{t_2} \xrightarrow{\mathbf{0}} \xrightarrow{t_3} \xrightarrow{\mathbf{2}} \lceil z \rceil$$

are feasible in Z, too.

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} \mathbf{1.9} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{4.2} \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z}$$

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 4.0 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\tau_1^*)} \lfloor z \rfloor} \qquad \underbrace{\left(m_\sigma, \begin{pmatrix} 1.9 \\ 1.4 \\ 1.4 \\ 1.4 \\ 4.2 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z} \qquad \underbrace{\left(m_\sigma, \begin{pmatrix} 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \\ 5.0 \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\tau_2^*)} \lceil z \rceil}$$

State Space Reduction

Example (continuation)

$$\underbrace{\left(m_\sigma, \begin{pmatrix} \mathbf{1.0} \\ \mathbf{1.0} \\ \mathbf{1.0} \\ \mathbf{1.0} \\ \mathbf{4.0} \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\tau_1^*)} \lfloor z \rfloor} \leq \underbrace{\left(m_\sigma, \begin{pmatrix} \mathbf{1.9} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{1.4} \\ \mathbf{4.2} \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\hat{\beta})} z} \leq \underbrace{\left(m_\sigma, \begin{pmatrix} \mathbf{2.0} \\ \mathbf{2.0} \\ \mathbf{2.0} \\ \mathbf{2.0} \\ \mathbf{5.0} \\ \sharp \end{pmatrix}\right)}_{z_0 \xrightarrow{\sigma(\tau_2^*)} \lceil z \rceil}$$

State Space Reduction

Corollary

- *Each feasible t-sequence σ in Z can be realized with an "integer" run.*

State Space Reduction

Corollary

- *Each feasible t-sequence σ in Z can be realized with an "integer" run.*
- *Each reachable p-marking in Z can be found using "integer" runs only.*

State Space Reduction

Corollary

- *Each feasible t-sequence σ in Z can be realized with an "integer" run.*
- *Each reachable p-marking in Z can be found using "integer" runs only.*
- *If z is reachable in Z, then $\lfloor z \rfloor$ and $\lceil z \rceil$ are reachable in Z, too.*

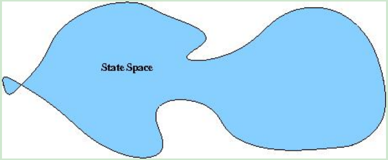
State Space Reduction

Corollary

- *Each feasible t-sequence σ in Z can be realized with an "integer" run.*
- *Each reachable p-marking in Z can be found using "integer" runs only.*
- *If z is reachable in Z, then $\lfloor z \rfloor$ and $\lceil z \rceil$ are reachable in Z, too.*
- *The length of the shortest and longest time path between two arbitrary p-markings are natural numbers.*

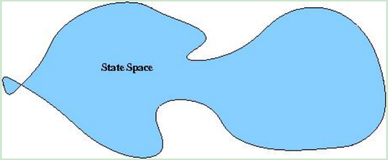
State Space Reduction

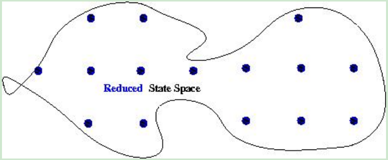
Example (State Space Reduction)



State Space Reduction

Example (State Space Reduction)





State Space Reduction

Theorem

Let Z be a FTPN.
The set of all reachable integer states in Z is finite

if and only if

the set of all reachable $p-$markings in Z is finite.

State Space Reduction

Theorem

Let Z be a FTPN.
The set of all reachable integer states in Z is finite

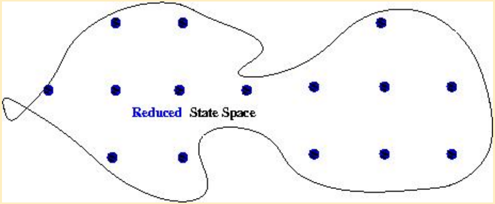
if and only if

the set of all reachable $p-$markings in Z is finite.

Remark: The theorem can be generalized for all TPNs (applying a further reduction).

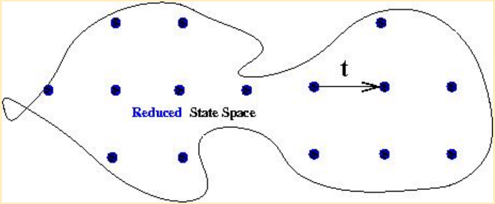
Reachability Graph

Definition (informal)



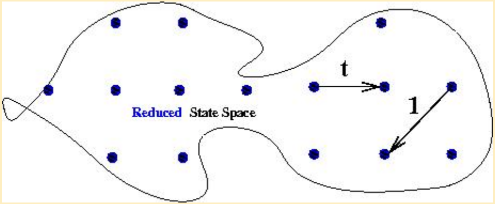
Reachability Graph

Definition (informal)



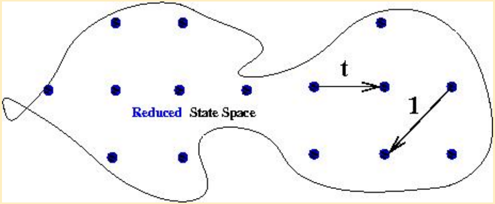
Reachability Graph

Definition (informal)



Reachability Graph

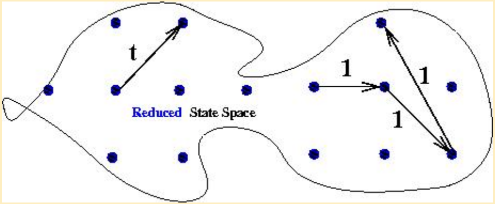
Definition (informal)



$\Longrightarrow$ The reachability graph is a directed graph.

Reachability Graph

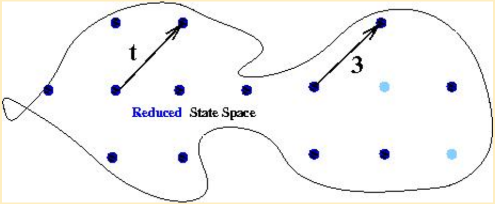
Definition (informal)



1st level of reduction

Reachability Graph

Definition (informal)

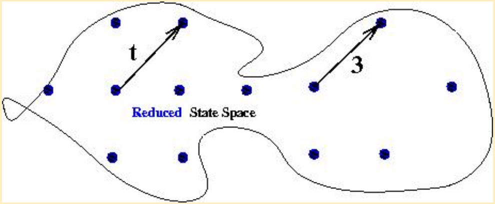


1st level of reduction

$\Longrightarrow$ The reachability graph is a weighted directed graph.

Reachability Graph

Definition (informal)

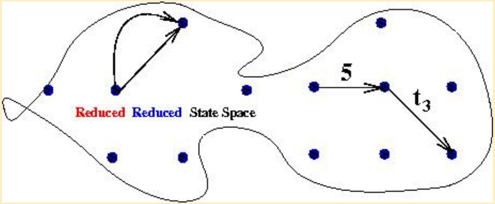


1st level of reduction

$\Longrightarrow$ The reachability graph is a weighted directed graph.

Reachability Graph

Definition (informal)

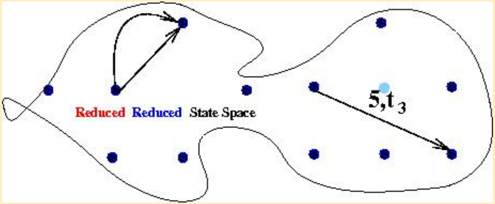


2nd level of reduction

$\Longrightarrow$ The reachability graph is a weighted directed graph.

Reachability Graph

Definition (informal)

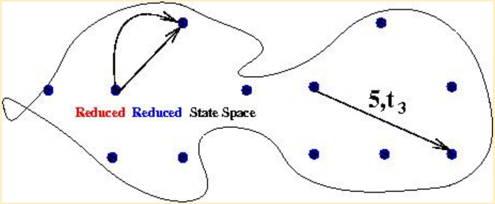


2nd level of reduction

$\Longrightarrow$ The reachability graph is a weighted directed graph.

Reachability Graph

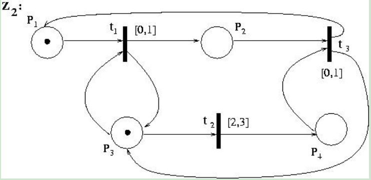
Definition (informal)

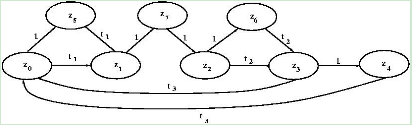


2nd level of reduction

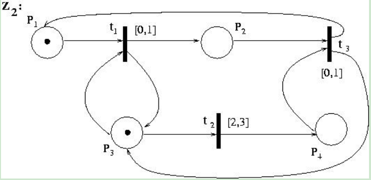
$\Longrightarrow$ The reachability graph is a weighted directed graph.

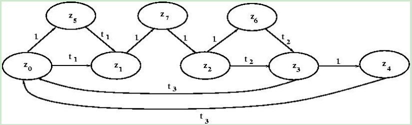
Example (The FTPN Z_2 and its reachability graph(s))

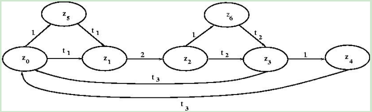




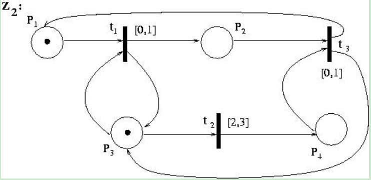
Example (The FTPN Z_2 and its reachability graph(s))

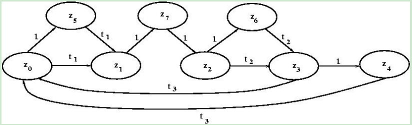


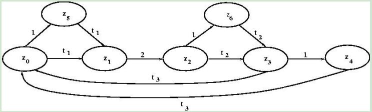


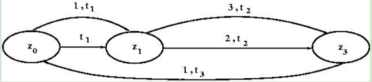


Example (The FTPN Z_2 and its reachability graph(s))

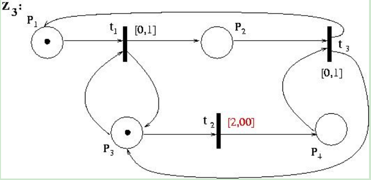


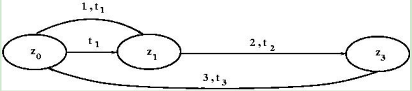






Example (The infinite TPN Z_3 and its reachability graph $RG(Z_3)$)





Arbitrary Time Petri Nets

Let $Z = (P, T, F, V, I, m_o)$ be an arbitrary TPN.
Then the following problems can be decided/computed **without knowledge** of its RG, amongst others:

Arbitrary Time Petri Nets

Result 1:

Input: The time function I is fixed,
σ is an arbitrary transition sequence.
Output: Feasibility of σ in Z?
Solution: Solve a linear system of inequalities in $\mathbb{R}_0^+$.
(polyn. running time)

Arbitrary Time Petri Nets

Result 2:

Input:	The time function I is not fixed,
	σ is an arbitrary transition sequence.
Output:	Feasibility of σ in Z for a fixed I?
Solution:	Solve a linear system of inequalities in $\mathbb{Q}_0^+$.
	(polyn. running time)

Arbitrary Time Petri Nets

Result 3:

Input:	The time function I is fixed, σ is an arbitrary transition sequence.
Output:	min / max-length of σ.
Solution:	Solve a linear program in $\mathbb{R}_0^+$. (Actually, the solution is in $\mathbb{N}$.) (polyn. running time)

Arbitrary Time Petri Nets

Result 4:

Input: The time function I is not fixed,
σ is an arbitrary transition sequence,
λ is an arbitrary real number.

Output: Existence of a fixed I and a run $\sigma(\tau)$ in Z and the length of $\sigma(\tau) \leq \lambda$?

Solution: Solve a system of linear equalities in $\mathbb{Q}_0^+$.
(polyn. running time)

Arbitrary Time Petri Nets

Result 5:

Input: The time function I is not fixed,
$\sigma_1 = (\sigma, t')$ is a arbitrary t-sequence and
$\sigma_2 = (\sigma, t'')$ is a arbitrary t-sequence.

Output: Existence of a fixed I so that σ_1 is feasible in Z and σ_2 is not feasible in Z?

Solution: Solve

$$\underbrace{\max\{< c', x >|\ A' \cdot x \leq b'\}}_{\text{linear program in } \mathbb{Q}_0^+} < \underbrace{\min\{< c'', x >|\ A'' \cdot x \leq b''\}}_{\text{linear program in } \mathbb{Q}_0^+}.$$

Bounded Time Petri Nets

Let $Z = (P, T, F, V, I, m_o)$ be a bounded TPN. Additionally the following problems can be decided/computed **with the knowledge** of its RG, **by means of prevalent methods of the graph theory,** amongst others:

Bounded Time Petri Nets

Result 6:

Input: z and z' - two states (in Z).

Output: – Is there a path between z and z' in $RG(Z)$?
– If yes, compute the path with the shortest time length.

Solution: By means of prevalent methods of the graph theory, e.g. Bellman-Ford algorithm (the running time is $\mathcal{O}(|V| \cdot |E|)$ and $RG(Z) = (V, E)$)

Bounded Time Petri Nets

Result 7:

Input: m and m' - two p–markings (in Z).

Output: – Is there a path between m and m'?
– If yes, compute the path with the shortest time length.

Solution: By means of prevalent methods of the graph theory, for computing all-pairs shortest paths.
The running time is polynomial, too.

Bounded Time Petri Nets

Definition

The **longest path** between two states (vertices in $RG(Z)$) z and z' is $lp(z, z')$ with

$$lp(z,z') := \begin{cases} \infty & \text{, if a cycle is reachable starting on } z \\ & \text{before reaching } z' \\ \max\limits_{\sigma(\tau)} \sum\limits_i \tau_i & \text{, else, where } z \xrightarrow{\sigma(\tau)} z' \end{cases}$$

Bounded Time Petri Nets

Result 8:

Input: z and z' - two states (in Z).

Output: – Is there a path between z and z' in $RG(Z)$?
– If yes, compute the path with the longest time length.

Solution: By means of prevalent methods of the graph theory, e.g. Bellman-Ford algorithm (polyn. running time). or by computing all strongly connected components of $RG(Z)$. (linear running time)

Bounded Time Petri Nets

Result 9:

Input: m and m' - two p-markings (in Z).

Output:
– Is there a path between m and m'?
– If yes, compute the path with the longest time length.

Solution: By means of prevalent methods of the graph theory, for computing all-pairs longest paths in the graph $RG(Z)$.

T-Invariants in an arbitrary Time Petri Nets

Definition

The transition sequence σ is a **feasible T-invariant** in a TPN Z if for each marking m in Z holds: $m \xrightarrow{\sigma} m$.

T-Invariants in an arbitrary Time Petri Nets

Result 10:

Input: A TPN Z.

Output:
– Is there a T–Invariance σ in Z?
– If yes, compute σ.

Solution:
– Solve the linear system of equations $C \cdot x = 0$ for $x \in \mathbb{N}$.
– Decide feasibility of a T-invariant σ with Parikh$(\sigma) = x$ for the Petri Net $S(Z)$.
– σ is feasible, then solve the linear system of inequalities B_σ in $\mathbb{R}_0^+$.

T-Invariants in an arbitrary Time Petri Nets

Remark: The reachability graph of a TPN is not used for computing the feasible T-invariants of Z

$$\Longrightarrow$$

feasible T-invariants for **unbounded** nets can be computed!

Conclusion

- The "integer–states" in a TPN are the supporters of the the net behaviour.

Conclusion

- The "integer–states" in a TPN are the supporters of the the net behaviour.

  $$\Longrightarrow$$

  Definition of a RG using the "integer–states".

Conclusion

- The "integer–states" in a TPN are the supporters of the the net behaviour.

  $$\Longrightarrow$$

  Definition of a RG using the "integer–states".
- The minimal and the maximal time length of a path between two markings in a TPN are natural numbers (if finite)

Conclusion

- The "integer–states" in a TPN are the supporters of the the net behaviour.

  $$\Longrightarrow$$

  Definition of a RG using the "integer–states".

- The minimal and the maximal time length of a path between two markings in a TPN are natural numbers (if finite)

  $$\Longrightarrow$$

  it can be computed in polynomial/linear time (with res. to the RG)

Conclusion

- The "integer–states" in a TPN are the supporters of the the net behaviour.

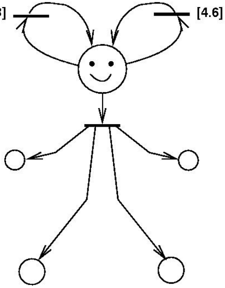
  $$\Longrightarrow$$

  Definition of a RG using the "integer–states".
- The minimal and the maximal time length of a path between two markings in a TPN are natural numbers (if finite)

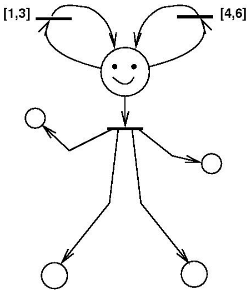
  $$\Longrightarrow$$

  it can be computed in polynomial/linear time (with res. to the RG)
- T–Invariances of an arbitrary TPN can be computed without knowledge of its RG.

Thank you!



Thank you!



Thank you!