Structural properties of Petri nets

dr. Tamás Bartha

dr. István Majzik

dr. András Pataricza

BME Department of Measurement and Information Systems
Recall: Dynamic properties

• Example: Model of a workflow (tasks + activities + resources)

• Properties analyzed
  – Does the system halt?  Deadlock
  – Can certain activities be performed?  Liveness
  – Do tasks overwhelm?  Boundedness
  – Can we return to the initial state?  Reversibility
  – Is there a processing loop?  Home state
  – Can activities be stopped?  Persistence
  – Is there an activity lacking resources?  Fairness

• Problem: Exploring a large state space
Recall: Analysis methods

Depth of the analysis:

- **Simulation**
- **Full exploration of the state space**
  - Analysis of reachability graph: Dynamic (behavioral) properties
  - Model checking
- **Analysis of the net structure**
  - Static analysis: Structural properties
  - Invariant analysis

- Traverse single trajectories
- Traverse all trajectories from a given initial state (exhaustive traversal)
- Properties independent from the initial state (hold for every initial state)
Main idea of structural analysis

• Can we state something without traversing / exploring the state space?
  – Based only on the structure (places, transitions, arcs)
  – Analysis independent from the initial state
  – In certain cases only approximate results!

• Approximate analysis is safe if it covers the real behavior
  – If no counterexample is found for the examined property (erroneous behavior): the property holds
  – If a counterexample is found: it may be spurious: It has to be verified with simulation and if it is spurious a new search has to be started
Structural properties

Properties of Petri nets independent from the initial state:

- Structural boundedness
- Controllability
- Conservativeness
  - Place invariant (P-invariant)
- Structural liveness
- Repetitiveness
- Consistence
  - Transition invariant (T-invariant)

Depending on the definition, the property must hold for

- either for all bounded initial marking,
- or some existing bounded initial marking
Recall: Describing the structure

- **Weighted incidence matrix**: $\mathbf{W} = [w(t, p)]$
- **Dimension**: $\tau \times \pi = |T| \times |P|$
- $w(t, p)$: Change in the number of tokens on $p$ when $t$ fires

\[
\begin{align*}
\mathbf{W} &= \mathbf{W}^+ - \mathbf{W}^- \\
\mathbf{W}^+ &= \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 4 & 1 \\
0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix} \\
\mathbf{W}^- &= \begin{bmatrix}
2 & 0 & 0 & 0 & 0 & 0 \\
1 & 3 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{align*}
\]
Recall: Describing the structure

\[ W = \begin{pmatrix} p_1 & p_2 & p_3 \\ t_1 & 2 & 1 & -1 \\ t_2 & 0 & 1 & -1 \\ t_3 & -1 & -1 & 1 \end{pmatrix} \]

\[ W^T = \begin{pmatrix} t_1 & t_2 & t_3 \\ p_1 & 2 & 0 & -1 \\ p_2 & 1 & 1 & -1 \\ p_3 & -1 & -1 & 1 \end{pmatrix} \]
Introducing the state equation

• Dynamics of Petri nets: change in the marking
  – Changes can be described by equations

• Precondition (for unambiguousness): pure Petri net
  – No transition exists that is both the input and output transition of the same place: \( \forall t \in T : \bullet t \cap t\bullet = \emptyset \)
  – This subsumes: No “self-loop”

    • Marking does not change after firing
      (0 element in the incidence matrix)

    • But has a role in enabling the transition
Firing sequence

- Firing sequence:
  \[ \tilde{\sigma} = \langle M_{i_0} t_{i_1} M_{i_1} \ldots t_{i_n} M_{i_n} \rangle = \langle t_{i_1} \ldots t_{i_n} \rangle \]

- Reachability of a state (marking):
  \[ M_{i_0} [\tilde{\sigma} > M_{i_n}] \]

- Enabledness of a firing sequence:
  - Transition \( t_{i,j} \) has enough tokens on input places \( p \in \bullet t_{i,j} \)
  \[ \forall t_{i,j} \in \tilde{\sigma}, \forall p \in \bullet t_{i,j} : M_{i_{j-1}}(p) \geq \mathbf{w}^-(p, t_{i,j}) = \mathbf{W}^{-t} \hat{e}_{i,j} \]
State equation

• **Change in the marking:**
  – When firing an enabled transition $t_j$
    • $w^-(p, t_j)$ tokens removed from each input place $p \in \bullet t_j$
    • $w^+(p, t_j)$ tokens are produced in each output place $p \in t_j\bullet$
  
  $$M_j = M_{j-1} - W^{-T} \bar{e}_j + W^{+T} \bar{e}_j = M_{j-1} + W^T \bar{e}_j$$

  – When firing an enabled firing sequence $\sigma$:
    • Marking changes by accumulating the firings:

  $$M_0 \left[ \bar{\sigma} > M_j \rightarrow M_j = M_0 + W^T \bar{\sigma}_T \right]$$

• **Firing count vector:** number of occurrences for each transition in the firing sequence
Deriving the state equation

\[ M_1 = M_0 + W^T \hat{e}_{t_1} \]

substituting \( M_1 \)

\[ M_2 = M_1 + W^T \hat{e}_{t_2} = M_0 + W^T \hat{e}_{t_1} + W^T \hat{e}_{t_2} \]

\[ M_{n+1} = M_n + W^T \hat{e}_{t_{n+1}} = M_0 + W^T \hat{e}_{t_1} + W^T \hat{e}_{t_2} + \ldots + W^T \hat{e}_{t_{n+1}} \]

\[ M_m = M_0 + W^T \hat{e}_{t_1} + W^T \hat{e}_{t_2} + \ldots + W^T \hat{e}_{t_m} = M_0 + W^T \sum_{i=1}^{m} \hat{e}_{t_i} \]

joined

\[ M_m = M_0 + W^T \tilde{\sigma}_T \implies M_m - M_0 = W^T \tilde{\sigma}_T \]
The firing count vector contains less information, than the firing sequence

- The order of firing is lost by only giving \( (0,2,2)^T \)!
- A non fireable sequence can be obtained from the state equation for a given \( M_0 \)
Example: State equation and reachability

- **State equation:**

\[ M_0 \left[ \bar{\sigma} > M_j \Rightarrow M_j - M_0 = W^T \bar{\sigma}_T \right] \]

- **Firing count vector:** \((1,0,1)^T\)

- But neither \(t_1\), nor \(t_3\) is enabled under the initial marking \((0,1,0)\)!
Transition and place invariants
Definition: Transition invariant (T-invariant)

The firing count vector $\sigma_T$ is a T-invariant, if its firing does not change the marking:

$$W^T \vec{\sigma}_T = 0$$

- Cycle in the state space: $M_i \left[ \vec{\sigma}_T > M_i \right]$
- The firing sequence $\sigma_T$ can be fired from state $M_i$ if

$$\forall t_{i,j} \in \vec{\sigma}, \forall p \in \{ \bullet t_{i,j} \} : m_{i_{j-1}}(p) \geq w^-(p, t_{i,j}) = W^{-T} \cdot \vec{e}_{i,j}$$

- Note: for each firing sequence $\sigma$ an initial marking $M_0$ exists, from which $\sigma$ can be fired

  - E.g. $M_0 \geq W^{-T} \vec{\sigma}$, the marking can have initially “as many” tokens, that the tokens produced by $\sigma$ are not needed
Example T-invariant

T-invariant: marking does not change after firing $t_1 - t_2$

Not a T-invariant: firing sequence $t_3 - t_1 - t_2$ cannot be repeated
Set of T-invariants

\[ W^T \sigma_T = 0 \]

Solutions of the homogeneous, linear system of equations

- Multiples of a solution are also solutions
  - If fireable, the loop can be traversed multiple times
- Sum of solutions is also a solution
  - If fireable, multiple loops can be combined
- Linear combination of solutions is also a solution

A basis can be found for the solutions

- Minimal set that can produce each solution
Minimal T-invariant

- Notation: basis of a firing sequence $\sigma$ is $\sup(\sigma)$:
  - Set of transitions $T' = \{t_i | \sigma_i > 0\}$ occurring in the sequence $\sigma$

- T-invariant $\sigma_T$ is minimal
  - If no T-invariant exists having a basis that is a proper subset of the basis of $\sigma_T$ or
  - if the subsets are equal, its firing counts are lower

\[
\forall \sigma_T^1: W^T \sigma_T^1 = 0 \Rightarrow \left( \sigma_T^1 \geq \sigma_T \right) \lor \left( \sup(\sigma_T) \not\subset \sup(\sigma_T^1) \right)
\]
Definition: Place invariant (P-invariant)

- A set of places marked by the non-negative weight vector $\mu_p$, where the weighted sum of tokens is constant:

$$\tilde{\mu}_p^T M = \text{constant}$$

- Number of tokens in a subset of places is constant (e.g. resources are not lost or introduced)

$$M = M_0 + W^T \tilde{\sigma}$$

$$\tilde{\mu}_p^T M = \tilde{\mu}_p^T M_0 + \tilde{\mu}_p^T W^T \tilde{\sigma}$$

$$\tilde{\mu}_p^T W^T \tilde{\sigma} = 0 \Rightarrow \tilde{\mu}_p^T W^T \forall \tilde{\sigma}$$

$$W \tilde{\mu}_p = 0$$
Example P-invariant

P-invariant for $p_1$, $p_2$, $p_3$:

Not a P-invariant:

Number of tokens increases

Weighted: $p_1 + p_2 + 2p_3 = \text{const.}$
Applications of invariants

- **Applications of T-invariants**
  - For a process model: cyclical behavior
  - Dynamic properties
    - Cyclically fireable $\rightarrow$ reversibility, home state
    - Can be fired later $\rightarrow$ liveness, deadlock freedom

- **Applications of P-invariants**
  - For a process model: constant resources
  - Dynamic properties
    - Tokens are not lost $\rightarrow$ liveness, deadlock freedom
    - Tokens are not produced $\rightarrow$ boundedness
Calculating invariants
Does the example have invariants?

- For a P-invariant: \( \mathbf{W} \cdot \mu_P = 0 \)
  
  \[
  \mathbf{W} = \begin{pmatrix}
  p_1 & p_2 & p_3 \\
  t_1 & 2 & 1 & -1 \\
  t_2 & 0 & 1 & -1 \\
  t_3 & -1 & -1 & 1
  \end{pmatrix}
  \]
  
  \( \mathbf{W} \cdot (0, 1, 1)^T = 0 \)

- For a T-invariant: \( \mathbf{W}^T \cdot \sigma_T = 0 \)
  
  \[
  \mathbf{W}^T = \begin{pmatrix}
  p_1 & p_2 & p_3 \\
  t_1 & 2 & 0 & -1 \\
  t_2 & 1 & 1 & -1 \\
  t_3 & -1 & -1 & 1
  \end{pmatrix}
  \]
  
  \( \mathbf{W}^T \cdot (1, 1, 2)^T = 0 \)
Example: Processor data transmission

- **Processor**
  - waiting (idle)
  - asking for bus grant
  - placing address to bus
  - placing data to bus

- **Bus(es)**
  - Idle (not used)
  - busy (processor/periphery)

- **Petri net**
  - \( n = 4 \) transitions
  - \( m = 6 \) places
P-invariants: Calculate by hand!

Four P-invariants can be found
Example: Incidence matrices

\[
W^- = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 \\
\end{bmatrix}
\]

\[
W^+ = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
\]
Example: Incidence matrices

\[ W = W^+ - W^- = \]

\[
\begin{pmatrix}
    p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\
    -1 & 1 & 0 & 0 & 0 & 0 \\
    0 & -1 & 1 & 0 & -1 & 1 \\
    0 & 0 & -1 & 1 & 0 & 0 \\
    1 & 0 & 0 & -1 & 1 & -1 \\
\end{pmatrix}
\]
Example: Incidence matrices

$$W^T = \begin{bmatrix} t_1 & t_2 & t_3 & t_4 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$
Martinez-Silva algorithm: Initialization

\( i \leftarrow 1 \)

\( T_i \leftarrow \{ t \in T \} \)

\( A \leftarrow W^T, \ D \leftarrow 1_n \quad // \quad n = |P| \)

\( Q_i \leftarrow [\mathbf{D} \mid \mathbf{A}] \quad // \) identity matrix and incidence matrix

\( L_p \leftarrow \) the \( p \)th row of \( Q_i \)

\( T_1 = \{ t_1, t_2, t_3, t_4 \} \)

\[
Q_1 = \begin{bmatrix}
e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \\
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1
\end{bmatrix}
\]

\( p_1 \)

\( p_2 \)

\( p_3 \)

\( p_4 \)

\( p_5 \)

\( p_6 \)
Martinez-Silva algorithm: Loop

while \( A_i \neq 0 \)

if \( t_j \in T_i \) // choose a column not yet examined

\( T_{i+1} \leftarrow T_i \backslash \{ t_j \} \)

\( L_{\text{delete}} \leftarrow \emptyset \)

\( Q_{i+1} \leftarrow Q_i \)

for all \( u, v : A_i(u, j) \neq 0 \land A_i(v, j) \neq 0 \land \exists \lambda_u, \lambda_v \in \mathbb{R}^+ : \lambda_u A_i(u, j) + \lambda_v A_i(v, j) = 0 \)

add row \( \lambda_u L_u + \lambda_v L_v \) to \( Q_{i+1} \)

\( L_{\text{delete}} \leftarrow L_{\text{delete}} \cup \{ L_u, L_v \} \)

end for

delete rows in \( L_{\text{delete}} \) from \( Q_{i+1} \)

\( i \leftarrow i + 1 \)

end while
Martinez-Silva algorithm: Step 1/1

\[
Q_1 = \begin{bmatrix}
e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \\
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\
\end{bmatrix}
\]
Martinez-Silva algorithm: Step 1/2

\[ Q_1 = \]

\[
\begin{bmatrix}
e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \\
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\
\end{bmatrix}
\]

\[ Q_{1'} = \]

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\
1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & p_{1+2} \\
\end{bmatrix}
\]
Martinez-Silva algorithm: Subresult 1

\[ Q_{1}'' = \begin{bmatrix}
  e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \\
  0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & p_3 \\
  0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & p_4 \\
  0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & p_5 \\
  0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & p_6 \\
  1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & p_{1+2}
\end{bmatrix} \]
Martinez-Silva algorithm: Step 2/1, 2/2

\[ Q_2 = \]

\[
\begin{bmatrix}
 e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\
 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 \\
 1 & 1 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
\begin{bmatrix}
 t_1 & t_2 & t_3 & t_4 \\
 0 & 1 & -1 & 0 \\
 0 & 0 & 1 & -1 \\
 0 & -1 & 0 & 1 \\
 0 & 1 & 0 & -1 \\
 0 & -1 & 0 & 1
\end{bmatrix}
\]

\[ Q_2' = \]

\[
\begin{bmatrix}
 e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\
 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 \\
 1 & 1 & 0 & 0 & 0 & 0 \\
 1 & 1 & 1 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 1 & 0 \\
 1 & 1 & 0 & 0 & 0 & 1 \\
 0 & 0 & 0 & 0 & 1 & 1 \\
 0 & 0 & 0 & 0 & 1 & 1
\end{bmatrix}
\]

\[
\begin{bmatrix}
 t_1 & t_2 & t_3 & t_4 \\
 1 & -1 & 0 & 0 \\
 0 & 1 & -1 & p_3 \\
 -1 & 0 & 1 & p_4 \\
 0 & 1 & 0 & -1 p_5 \\
 0 & 1 & 0 & -1 p_6 \\
 0 & -1 & 0 & 1 p_{1+2} \\
 1 & 1 & 1 & 0 p_{1+2+3} \\
 0 & 0 & 1 & 0 p_{3+5} \\
 1 & 1 & 0 & 0 p_{1+2+6} \\
 0 & 0 & 0 & 1 p_{5+6}
\end{bmatrix}
\]
Martinez-Silva algorithm: Subresult 2

\[
Q_2'' = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & p_4 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & p_{1+2+3} \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 & p_{3+5} \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & p_{1+2+6} \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & p_{5+6}
\end{bmatrix}
\]
Martinez-Silva algorithm: Step 3/1, 3/2

\[ Q_3 = \begin{bmatrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \ p_4 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \ p_{1+2+3} \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \ p_{3+5} \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \ p_{1+2+6} \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \ p_{5+6} \end{bmatrix} \]

\[ Q_{3'} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \ p_4 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \ p_{1+2+3} \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \ p_{3+5} \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \ p_{1+2+6} \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \ p_{5+6} \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \ p_{1+2+3+4} \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \ p_{3+4+5} \end{bmatrix} \]
Martinez-Silva algorithm: Final results

- **Invariants:**
  - Coefficients in the rows of matrix $D_m$ in the final matrix $Q_m = [D_m|0]$

- **Resulting P-invariants:**
  1. $m(p_1)+m(p_2)+m(p_6) = 1$
  2. $m(p_5)+m(p_6) = 1$
  3. $m(p_1)+m(p_2)+m(p_3)+m(p_4) = 1$
  4. $m(p_3)+m(p_4)+m(p_5) = 1$

- Sum of tokens can be determined from the initial marking
Example: Calculating T-invariants

Start

<table>
<thead>
<tr>
<th>t₁</th>
<th>p₁</th>
<th>p₂</th>
<th>p₃</th>
<th>p₄</th>
<th>p₅</th>
<th>s₁₁</th>
<th>s₁₂</th>
<th>s₁₃</th>
<th>s₁₄</th>
<th>s₁₅</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>1</td>
<td>-2</td>
<td>0</td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t₂</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>-2</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t₃</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t₄</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>-2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>t₅</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Step 1

(work with 5th column)

| 1  | 1  | 0  | 0  | 0  | 2  | -1  | -2  | -1  | 0   | (11+12) |
| 0  | 1  | 1  | 0  | 0  | 4  | -2  | -1  | -2  | 0   | (12+13) |
| 1  | 0  | 0  | 0  | 1  | 0  | 0   | -1  | 0   | 0   | (11+15) |
| 0  | 0  | 1  | 0  | 1  | 2  | -1  | 0   | -1  | 0   | (13+15) |

(delete and reorder)

Before step 2

| 0  | 0  | 0  | 1  | 0  | -2 | 1   | 2   | 1   | 0   | s₂₁ |
| 1  | 1  | 0  | 0  | 0  | 2  | -1  | -2  | -1  | 0   | s₂₂ |
| 0  | 1  | 1  | 0  | 0  | 4  | -2  | -1  | -2  | 0   | s₂₃ |
| 1  | 0  | 0  | 0  | 1  | 0  | 0   | -1  | 0   | 0   | s₂₄ |
| 0  | 0  | 1  | 0  | 1  | 2  | -1  | 0   | -1  | 0   | s₂₅ |

Step 2

(work with 4th column)

| 1  | 1  | 0  | 1  | 0  | 0  | 0   | 0   | 0   | 0   | (21+22) |
| 0  | 0  | 1  | 1  | 1  | 0  | 0   | 2   | 0   | 0   | (21+25) |
| 0  | 1  | 1  | 2  | 0  | 0  | 0   | 3   | 0   | 0   | (2*21+23) |

(delete and reorder)

Before step 3

| 1  | 0  | 0  | 0  | 1  | 0  | 0   | -1  | 0   | 0   | s₃₁ |
| 1  | 1  | 0  | 1  | 0  | 0  | 0   | 0   | 0   | 0   | s₃₂ |
| 0  | 0  | 1  | 1  | 1  | 0  | 0   | 2   | 0   | 0   | s₃₃ |
| 0  | 1  | 1  | 2  | 0  | 0  | 0   | 3   | 0   | 0   | s₃₄ |

Step 3

(work with 3rd column)

| 2  | 0  | 1  | 1  | 3  | 0   | (2*31+33) |
| 3  | 1  | 1  | 2  | 3  | 0   | (3*31+34) |

39
Structural properties of Petri nets
Structural liveness, structural boundedness

- A Petri net $N$ is structurally live, if there exists a live initial marking $M_0$ for $N$
  - A Petri net is live, if it is L4-live, i.e., each transition $t \in T$ is L4-live
    - A transition is L4-live: can be fired at least once in some firing sequence from any reachable state

- A Petri net $N$ is structurally bounded, if it is bounded for all bounded initial markings $M_0$
Controllability

- A Petri net $N$ is completely controllable, if for all bounded initial marking $M_0$ any marking is reachable from any other marking, i.e.,

$$\forall M_i, M_j : M_i, M_j \in R(N, M_0) \Rightarrow M_i \in R(N, M_j) \land M_j \in R(N, M_i)$$
Conservativeness

• A Petri net $\mathcal{N}$ is conservative, if there exists a positive integer weight $\mu_p$ for every place $p \in P$ in every bounded $M_0$ and $M \in R(\mathcal{N}, M_0)$ such that:

$$M \hat{\mu} = M_0 \hat{\mu} = \text{constant}$$

  – Example: For each initial marking, each place in each reachable marking is part of a P-invariant

• Partially conservative, if the above only holds for some places.

  – Example: For each initial marking, some places in each reachable marking is part of a P-invariants
Repetitiveness

• A Petri net $N$ is repetitive, if an initial marking $M_0$ and a firing sequence $\sigma$ from $M_0$ exists, such that every transition $t \in T$ occurs infinitely often in $\sigma$.
  – Example: An initial marking exists with a returning firing sequence (loop) containing every transition

• Partially repetitive, if the above only holds for some transitions.
  – Example: An initial marking exists with a returning firing sequence (loop) containing some transitions
Consistency

- A Petri net $\mathcal{N}$ is consistent, if an initial marking $M_0$ and a firing sequence $\sigma$ from $M_0$ to $M_0$ exists, such that every transition $t \in T$ occurs at least once in $\sigma$.

- Partially consistent, if the above only holds for some transitions.
Structural B-fairness

- Two transitions are structurally B-fair, if for all initial markings $M_0$ the two transitions are B-fair
  - Two transitions are B-fair: One of them can fire only a bounded number of times without firing the other

- A Petri net $N$ is structurally B-fair, if for all initial markings $M_0$ the net is B-fair
  - A Petri net $(N, M_0)$ is B-fair, if any two transitions are in a B-fair relationship
  - Structural B-fair relation $\iff$ B-fair relation
B-fair, but not structurally B-fair net

B-fair $M_0$

Not B-fair $M_0$
<table>
<thead>
<tr>
<th>Property</th>
<th>Necessary and sufficient condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB Structurally bounded</td>
<td>$\exists \tilde{\mu} &gt; 0, W \tilde{\mu} \leq 0$ (or $\exists \tilde{\sigma} &gt; 0, W^T \tilde{\sigma} \geq 0$)</td>
</tr>
<tr>
<td>CN Conservative</td>
<td>$\exists \tilde{\mu} &gt; 0, W \tilde{\mu} = 0$ (or $\exists \tilde{\sigma}, W^T \tilde{\sigma} \geq 0$)</td>
</tr>
<tr>
<td>PCN Partially conservative</td>
<td>$\exists \tilde{\mu} \geq 0, W \tilde{\mu} = 0$</td>
</tr>
<tr>
<td>RP Repetitive</td>
<td>$\exists \tilde{\sigma} &gt; 0, W^T \tilde{\sigma} \geq 0$</td>
</tr>
<tr>
<td>PRP Partially repetitive</td>
<td>$\exists \tilde{\sigma} \geq 0, W^T \tilde{\sigma} \geq 0$</td>
</tr>
<tr>
<td>CS Consistent</td>
<td>$\exists \tilde{\sigma} &gt; 0, W^T \tilde{\sigma} = 0$ (or $\exists \tilde{\mu}, W \tilde{\mu} \geq 0$)</td>
</tr>
<tr>
<td>PCS Partially consistent</td>
<td>$\exists \tilde{\sigma} \geq 0, W^T \tilde{\sigma} = 0$</td>
</tr>
</tbody>
</table>
**Other properties**

<table>
<thead>
<tr>
<th>If ...</th>
<th>Then ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{N}$ structurally bounded and structurally live</td>
<td>$\mathcal{N}$ is conservative and consistent.</td>
</tr>
<tr>
<td>$\exists \bar{\mu} \geq 0, W\bar{\mu} \leq 0$</td>
<td>A non-live $M_0$ exists for $\mathcal{N}$. $\mathcal{N}$ is not consistent.</td>
</tr>
<tr>
<td>$\exists \bar{\mu} \geq 0, W\bar{\mu} \geq 0$</td>
<td>$(\mathcal{N}, M_0)$ is not bounded with live $M_0$. $\mathcal{N}$ is not consistent.</td>
</tr>
<tr>
<td>$\exists \bar{\sigma} \geq 0, W^T\bar{\sigma} \leq 0$</td>
<td>A non-live $M_0$ exists for structurally bounded $\mathcal{N}$. $\mathcal{N}$ is not consistent.</td>
</tr>
<tr>
<td>$\exists \bar{\sigma} \geq 0, W^T\bar{\sigma} \geq 0$</td>
<td>$\mathcal{N}$ is not structurally bounded. $\mathcal{N}$ not conservative.</td>
</tr>
</tbody>
</table>