3.5 Komplexität nebenläufiger Systeme

3.5.1 Überdeckungsgraph

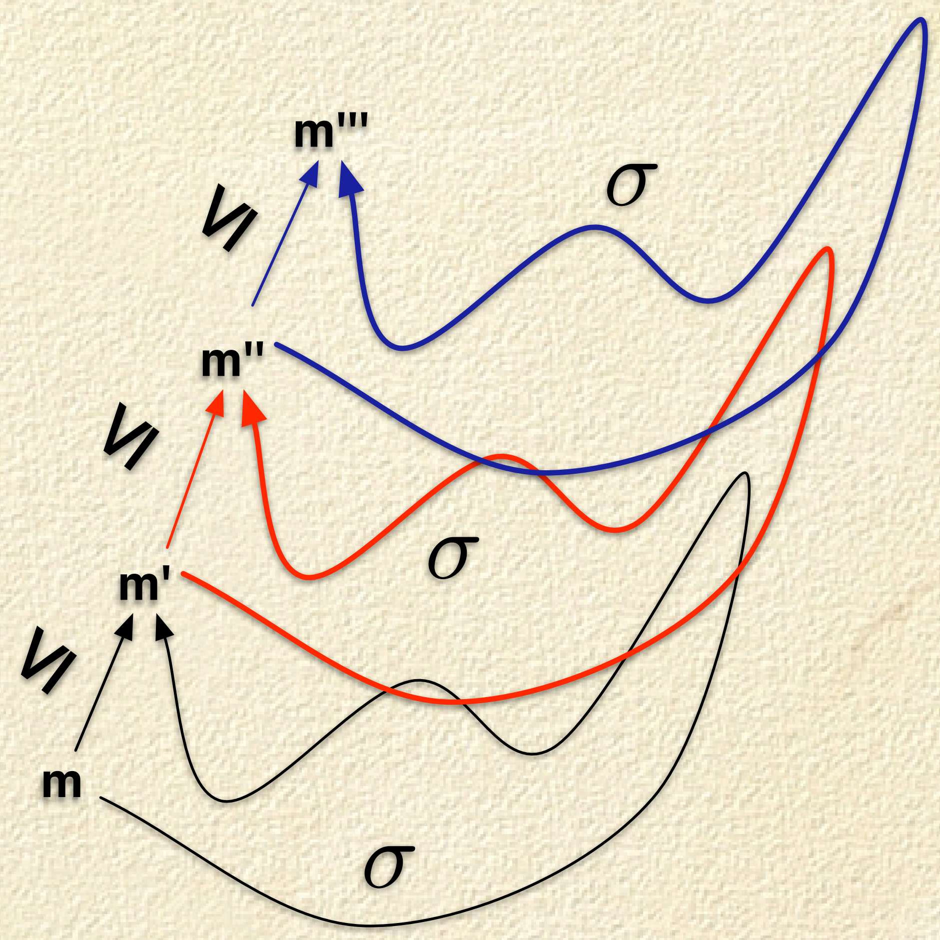
3.5.2 Turing-Mächtigkeit

3.5.3 Komplexität

3.5.1 Überdeckungsgraph

a) $\exists \sigma \in T^* : \mathbf{m} \xrightarrow{\sigma} \mathbf{m}'$

b) $\mathbf{m} \lneq \mathbf{m}'$



Definition 3.22 *Es sei* $\mathbb{N}_\omega := \mathbb{N} \cup \{\omega\}$ *zusammen mit folgenden Rechenregeln:*

$$\forall n \in \mathbb{N} : \omega > n;$$

$$\forall n \in \mathbb{N}_\omega : \omega + n = \omega - n := \omega;$$

$$\forall n \in \mathbb{N}\backslash\{0\} : n \cdot \omega = \omega \cdot n := \omega; 0 \cdot \omega = \omega \cdot 0 := 0.$$

Ein Vektor $\mathbf{m} \in \mathbb{N}_\omega^P$ *wird* Pseudomarkierung *genannt, wenn in ihm das Symbol* ω *vorkommt, und für diese wird die Schaltregel formal übernommen. Eine Pseudomarkierung* $\mathbf{m}$ *entspricht einer gewöhnlichen (Teil-) Markierung auf den Plätzen* s *mit* $\mathbf{m}(s) \neq \omega$*, wobei die mit* ω *besetzten Komponenten beliebig sind und unberücksichtigt bleiben.*

Für Vektoren $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{Z}^r$ *seien die Operatoren* $+, -, =$ $\leq$ *jeweils komponentenweise erklärt, d.h.,* $\mathbf{m}_1 \leq \mathbf{m}_2$*, falls* $\forall s \in S : \mathbf{m}_1(s) \leq \mathbf{m}_2(s)$*. Lediglich* $\mathbf{m}_1 < \mathbf{m}_2$ *steht für* $(\mathbf{m}_1 \leq \mathbf{m}_2$ *und* $\mathbf{m}_1 \neq \mathbf{m}_2)$*.*

$$\begin{pmatrix} 0 \\ 20 \\ 1 \end{pmatrix} \leq \begin{pmatrix} 0 \\ \omega \\ 2 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 20 \\ 1 \end{pmatrix} < \begin{pmatrix} 0 \\ \omega \\ 2 \end{pmatrix}$$

$X(\mathbf{m}')$

$$\mathbf{m}_2'' = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 3 \end{pmatrix} \qquad \mathbf{m}_3'' = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 1 \end{pmatrix}$$

$$\mathbf{m}_1'' = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 3 \end{pmatrix}$$

$$X(\mathbf{m}') := \{\mathbf{m}'' \in V \,|\, \mathbf{m}'' \leq \mathbf{m}' \text{ und } \mathbf{m}'' \xrightarrow{*} \mathbf{m}\}$$

$$\mathbf{m}$$

$$\downarrow$$

$$\mathbf{m}' = \cancel{\begin{pmatrix} 1 \\ 2 \\ 1 \\ 3 \end{pmatrix}} \quad \begin{pmatrix} \omega \\ \omega \\ 1 \\ \omega \end{pmatrix}$$

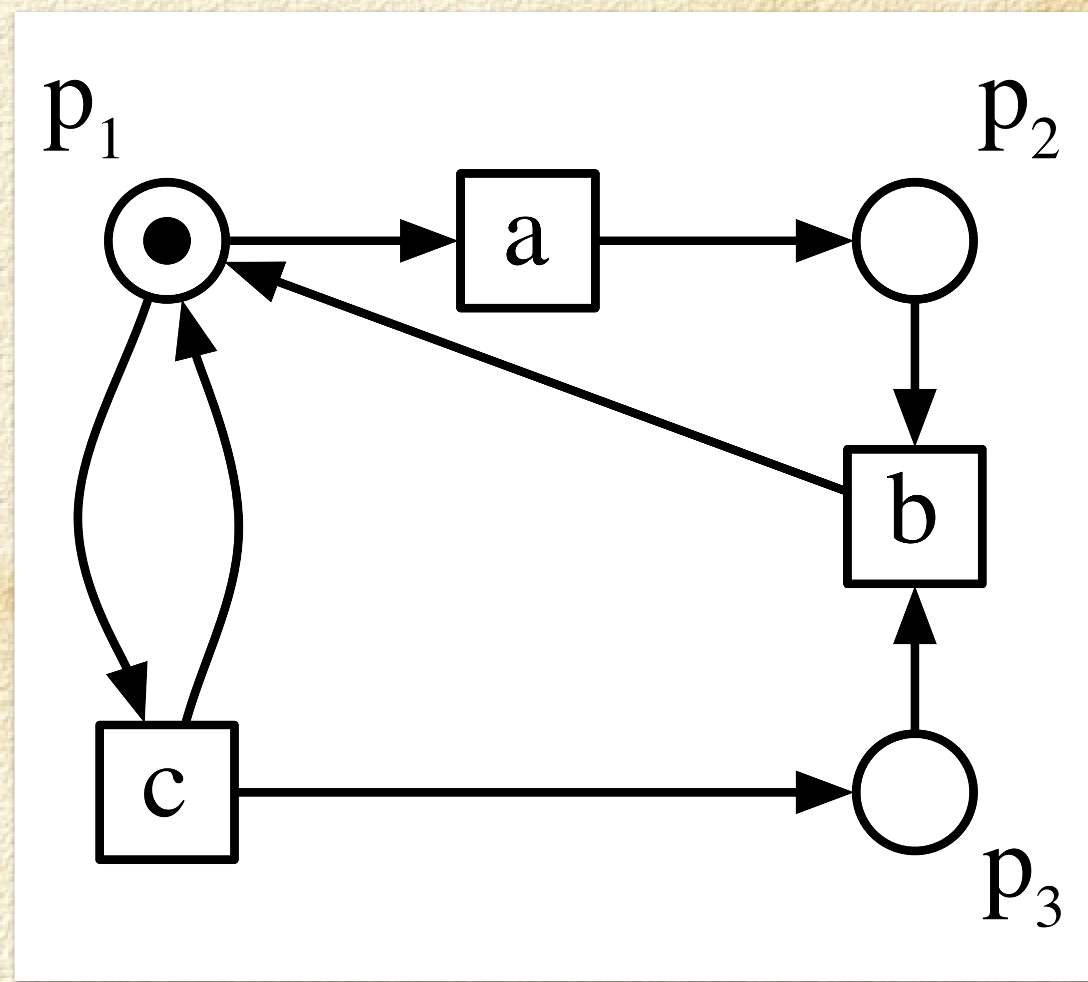
Algorithmus 3.4 (Berechnung des Überdeckungsgraph)

Input - Das P/T-Netz $\mathcal{N} = \langle P, T, F, W, \mathbf{m}_0 \rangle$
Output - Der Überdeckungsgraph Graph $G(\mathcal{N}) = (V, E)$.

1. Initialisiere $G(\mathcal{N}) = (\{\mathbf{m_0}\}, \emptyset)$; $\mathbf{m_0}$ sei ungefärbt;
2. **while** Es gibt ungefärbte Knoten in V **do**
   - 2.1 Wähle einen ungefärbte Knoten $\mathbf{m} \in V$ und färbe ihn.
   - 2.2 **for** Für jede in $\mathbf{m}$ aktivierte Transition t **do**
     - 2.2.1 Berechne $\mathbf{m}'$ mit $\mathbf{m} \xrightarrow{t} \mathbf{m}'$ und $X(\mathbf{m}') := \{\mathbf{m}'' \in V | \mathbf{m}'' \leq \mathbf{m}' \text{ und } \mathbf{m}'' \xrightarrow{*} \mathbf{m}'\}$;
     - 2.2.2 **if** $X(\mathbf{m}') \neq \emptyset$ **then** $\quad \mathbf{m}_1(p) := \begin{cases} \omega, & \exists \mathbf{m}'' \in X(\mathbf{m}') : \mathbf{m}''(p) < \mathbf{m}'(p) \\ \mathbf{m}'(p), & \text{sonst.} \end{cases}$

       else $\mathbf{m}_1 := \mathbf{m}'$;
     - 2.2.3 **if** $\mathbf{m}_1 \notin V$

       then $V := V \cup \{\mathbf{m}_1\}$, wobei $\mathbf{m}_1$ ein ungefärbter Knoten sei. ;
     - 2.2.4 $E := E \cup \{\langle \mathbf{m}, t, \mathbf{m}_1 \rangle\}$;
3. Der Algorithmus terminiert mit Ergebnis. ($G(\mathcal{N})$ ist der Überdeckungsgraph.)



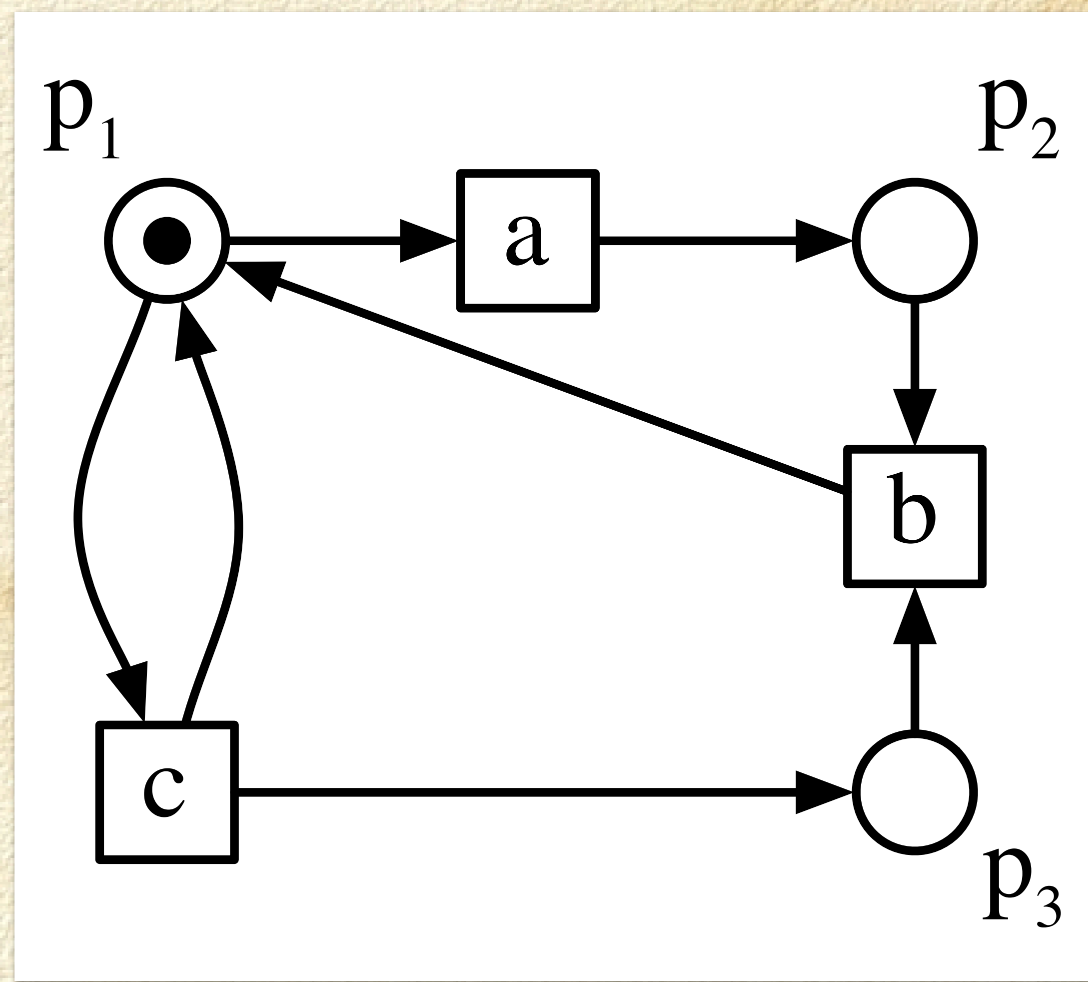
$$\begin{array}{cccc} & a & b & c \\ & \multicolumn{3}{c}{\left(\begin{array}{ccc} -1 & 1 & 0 \\ 1 & -1 & 0 \\ 0 & -1 & 1 \end{array}\right)} \end{array}$$

$$\begin{tikzcd} & (1,0,0) \arrow[dl, "a"'] \arrow[dr, "c"] & \\ (0,1,0) & & \cancel{(1,0,1)} \\ & & (1,0,\omega) \end{tikzcd}$$

ω? ω?

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$



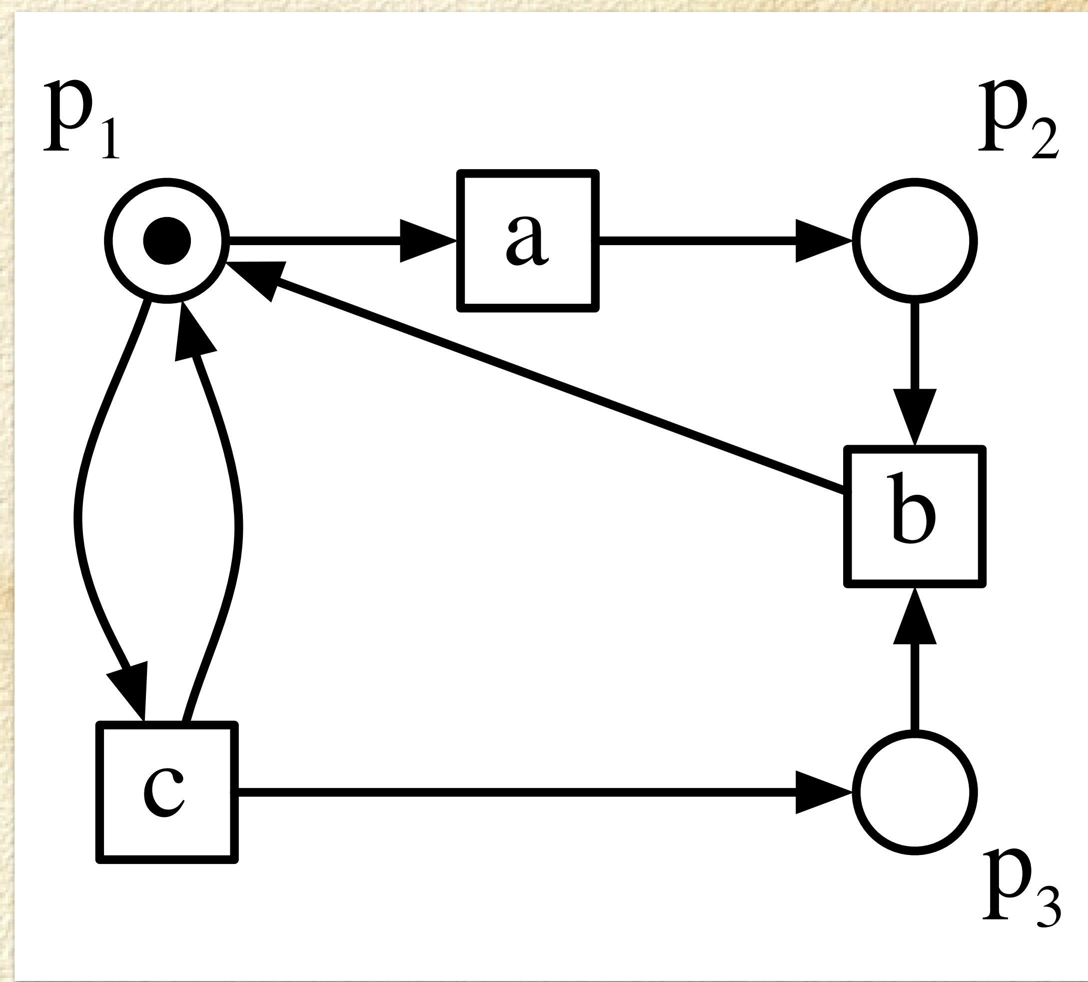
$$
\begin{array}{ccc}
\color{red}{a} & \color{red}{b} & \color{red}{c} \\
\end{array}
$$

$$
\begin{pmatrix}
-1 & 1 & 0 \\
1 & -1 & 0 \\
0 & -1 & 1
\end{pmatrix}
$$

$$
\begin{pmatrix} 1 \\ 0 \\ \omega \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ \omega \end{pmatrix}
$$

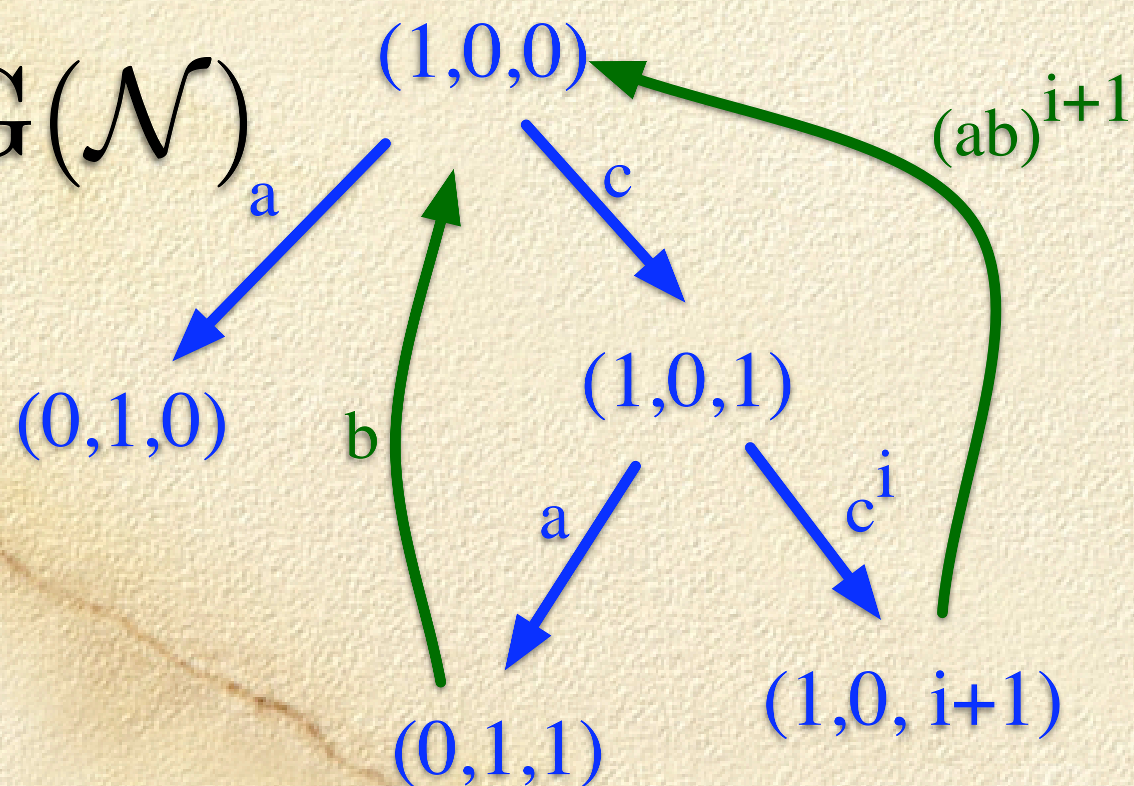
$$
\begin{pmatrix} 1 \\ 0 \\ \omega \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \omega \end{pmatrix}
$$

$$
\begin{tikzcd}
 & (1,0,0) \arrow[dl, "a"'] \arrow[dr, "c"] & & \\
(0,1,0) & & \begin{matrix}\cancel{(1,0,1)} \\ (1,0,\omega)\end{matrix} \arrow[dl, "a"'] \arrow[dr, "c"] & \\
 & (0,1,\omega) & & (1,0,\omega)
\end{tikzcd}
$$

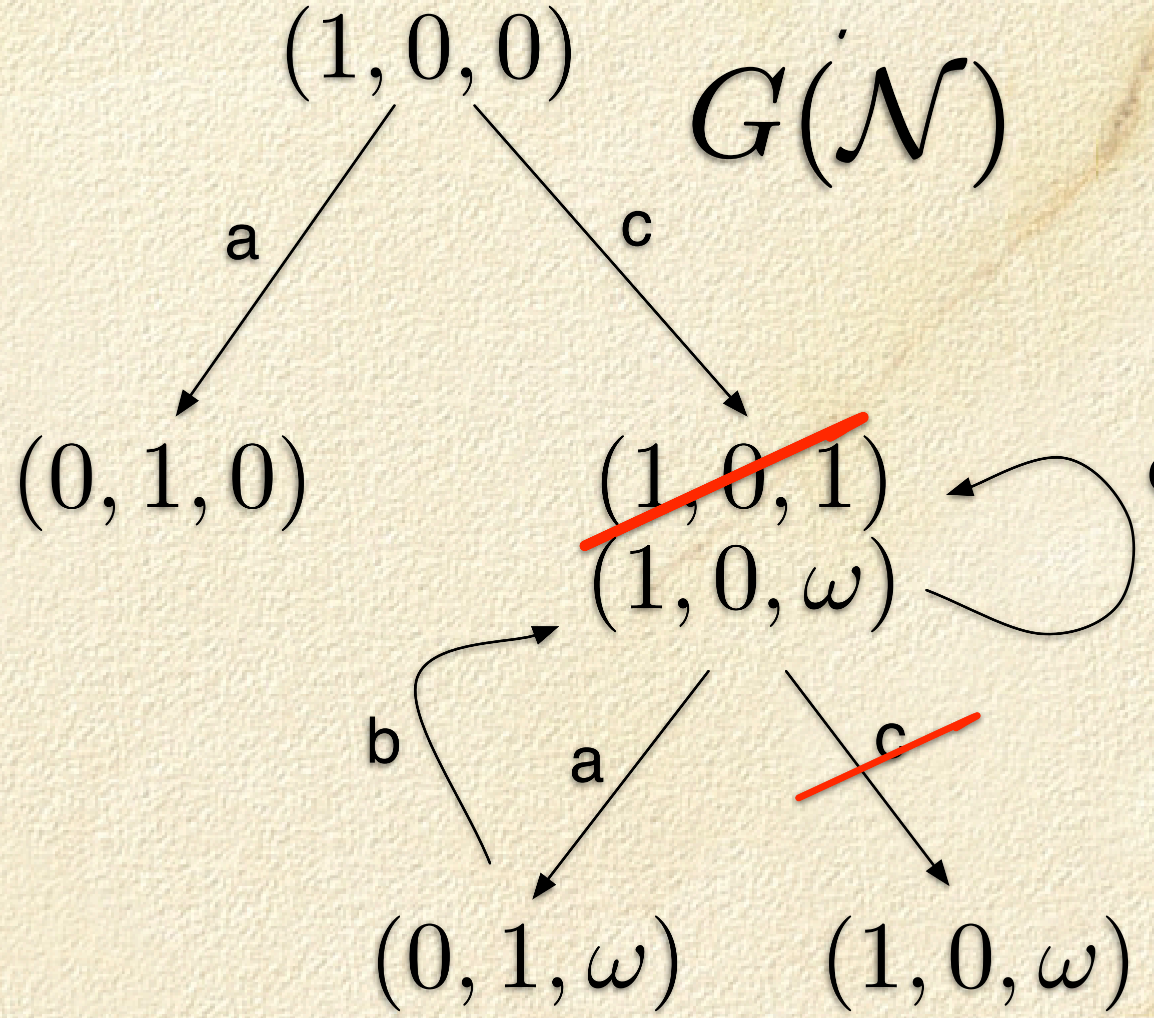


$$\begin{array}{ccc} \color{red}{a} & \color{red}{b} & \color{red}{c} \\ \end{array}$$
$$\begin{pmatrix} -1 & 1 & 0 \\ 1 & -1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

$RG(\mathcal{N})$



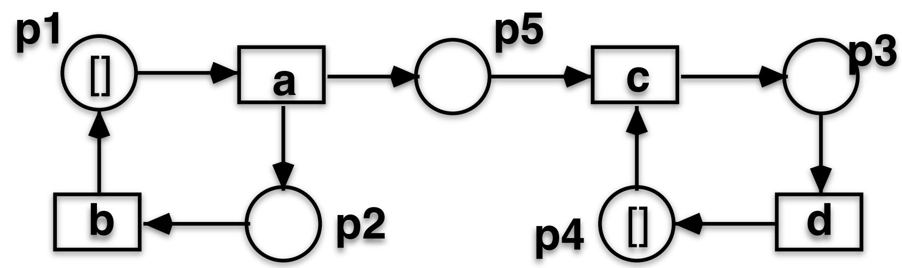
$G(\mathcal{N})$



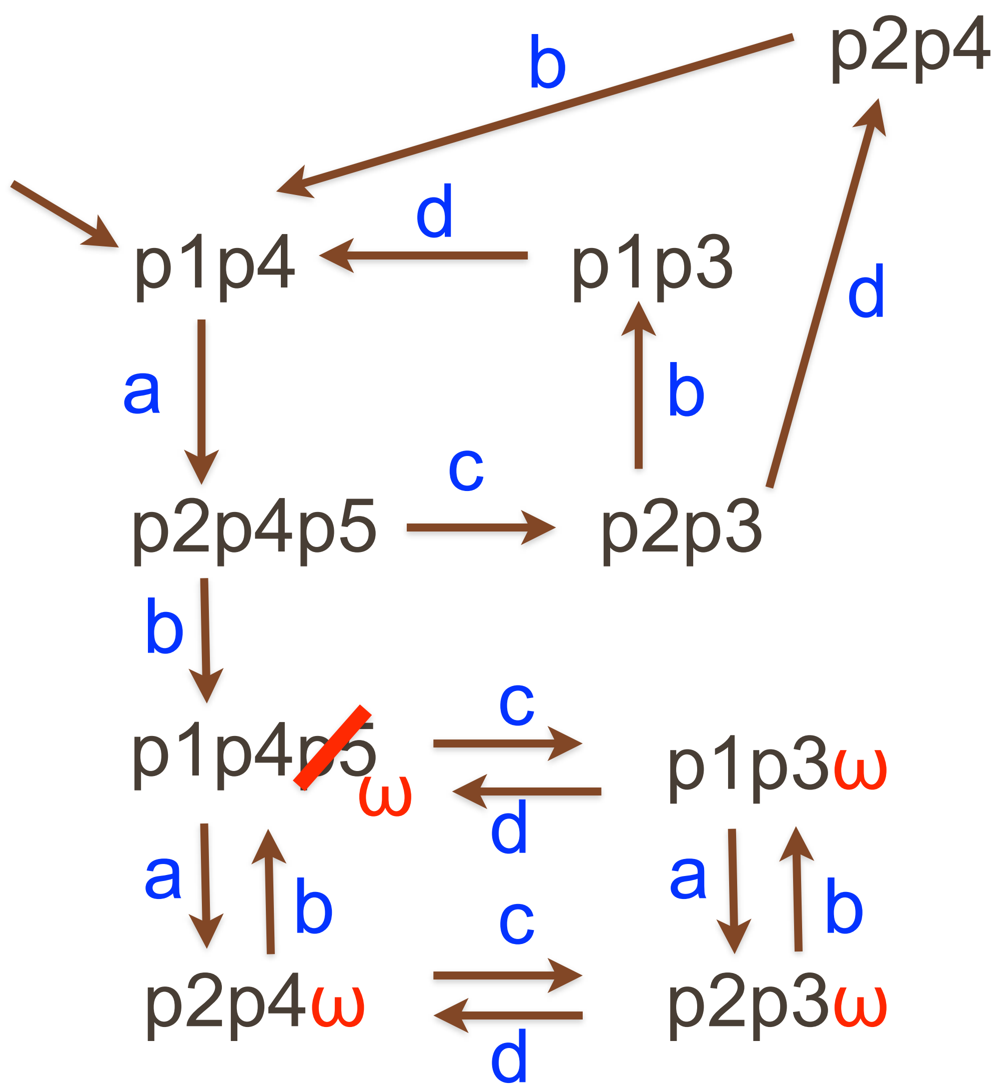
Algorithmus 3.4 (Berechnung des Überdeckungsgraph)

Input - Das P/T-Netz $\mathcal{N} = \langle P, T, F, W, \mathbf{m}_0 \rangle$
Output - Der Überdeckungsgraph Graph $G(\mathcal{N}) = (V, E)$.

1. Initialisiere $G(\mathcal{N}) = (\{\mathbf{m_0}\}, \emptyset)$; $\mathbf{m_0}$ sei ungefärbt;
2. **while** Es gibt ungefärbte Knoten in V **do**
   2.1 Wähle einen ungefärbte Knoten $\mathbf{m} \in V$ und färbe ihn.
   2.2 **for** Für jede in $\mathbf{m}$ aktivierte Transition t **do**
   2.2.1 Berechne $\mathbf{m}'$ mit $\mathbf{m} \xrightarrow{t} \mathbf{m}'$ und $X(\mathbf{m}') := \{\mathbf{m}'' \in V | \mathbf{m}'' \leq \mathbf{m}'$ und $\mathbf{m}'' \xrightarrow{*} \mathbf{m}'\}$;
   2.2.2 **if** $X(\mathbf{m}') \neq \emptyset$ **then** $\mathbf{m}_1(p) := \begin{cases} \omega, & \exists \mathbf{m}'' \in X(\mathbf{m}') : \mathbf{m}''(p) < \mathbf{m}'(p) \\ \mathbf{m}'(p), & \text{sonst.} \end{cases}$
   else $\mathbf{m}_1 := \mathbf{m}'$;
   2.2.3 **if** $\mathbf{m}_1 \notin V$
   then $V := V \cup \{\mathbf{m}_1\}$, wobei $\mathbf{m}_1$ ein ungefärbter Knoten sei. ;
   2.2.4 $E := E \cup \{\langle \mathbf{m}, t, \mathbf{m}_1 \rangle\}$;
3. Der Algorithmus terminiert mit Ergebnis. ($G(\mathcal{N})$ ist der Überdeckungsgraph.)

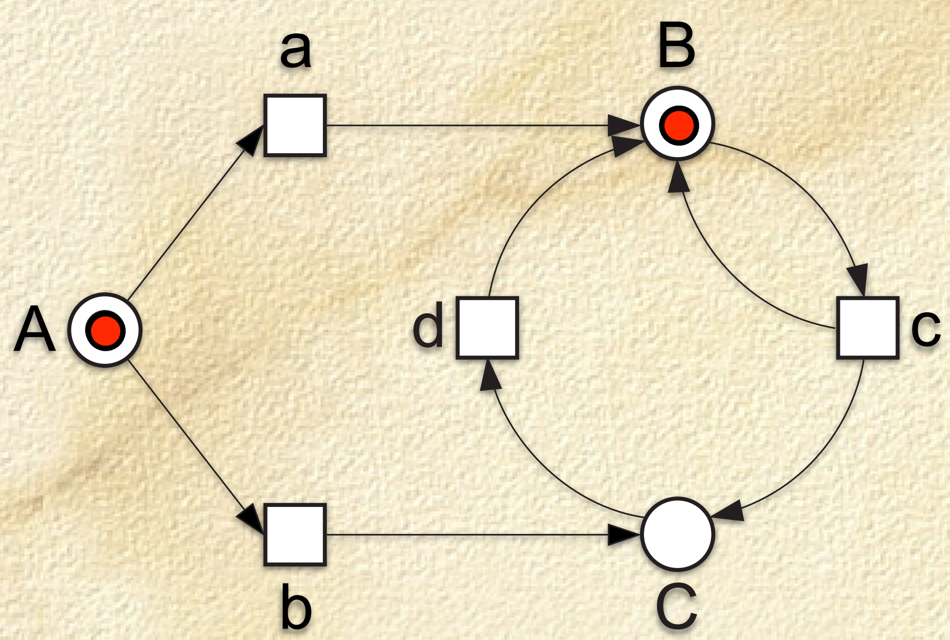


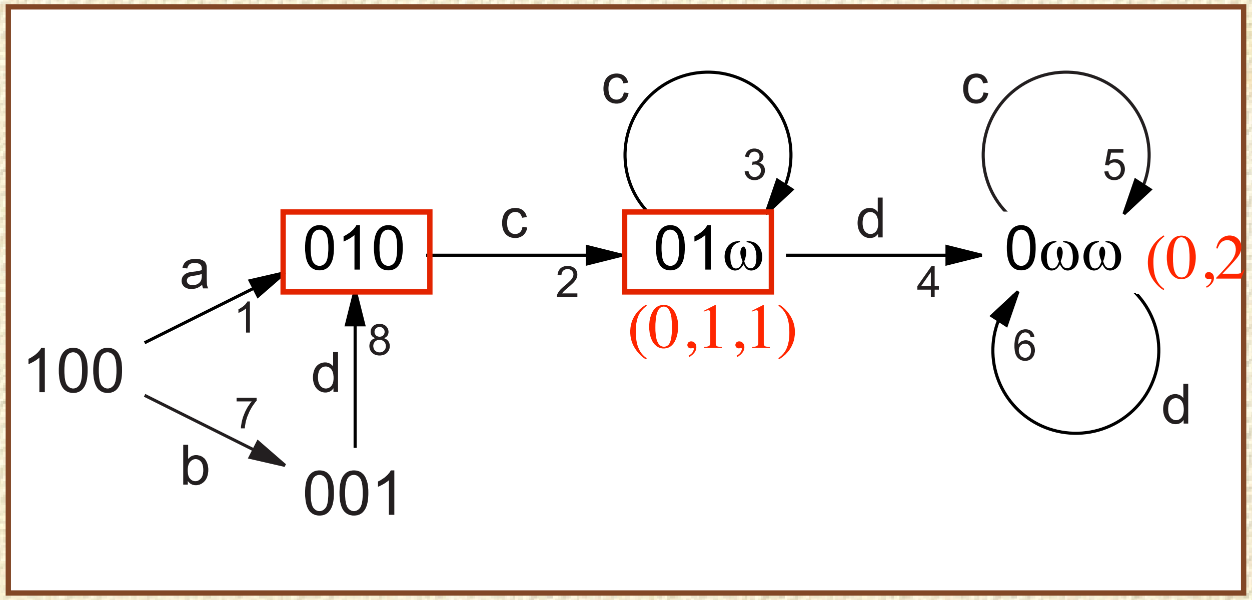
4. Konstruiere den Überdeckungsgraphen.

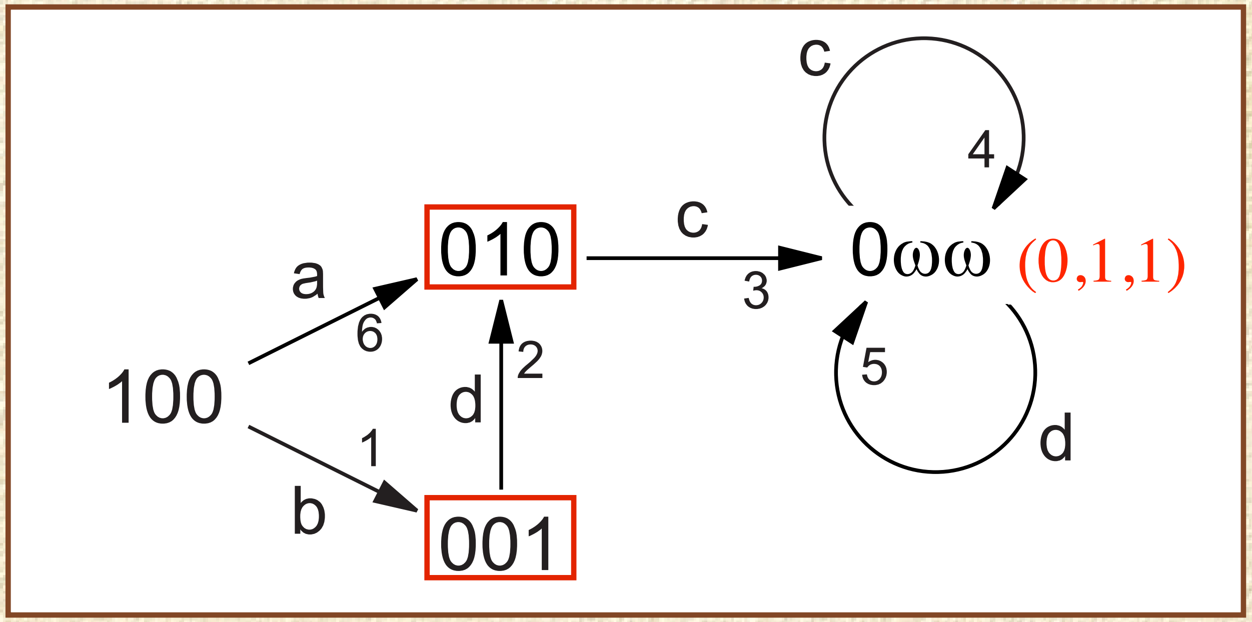


5. Beschreibe mit eigenen Worten den Hinweis zur Aufgabe 8.4.

der Überdeckungsgraph oder ein Überdeckungsgraph?







$X(\mathbf{m}')$

5. Beschreibe mit eigenen Worten den Hinweis zur Aufgabe 8.4.

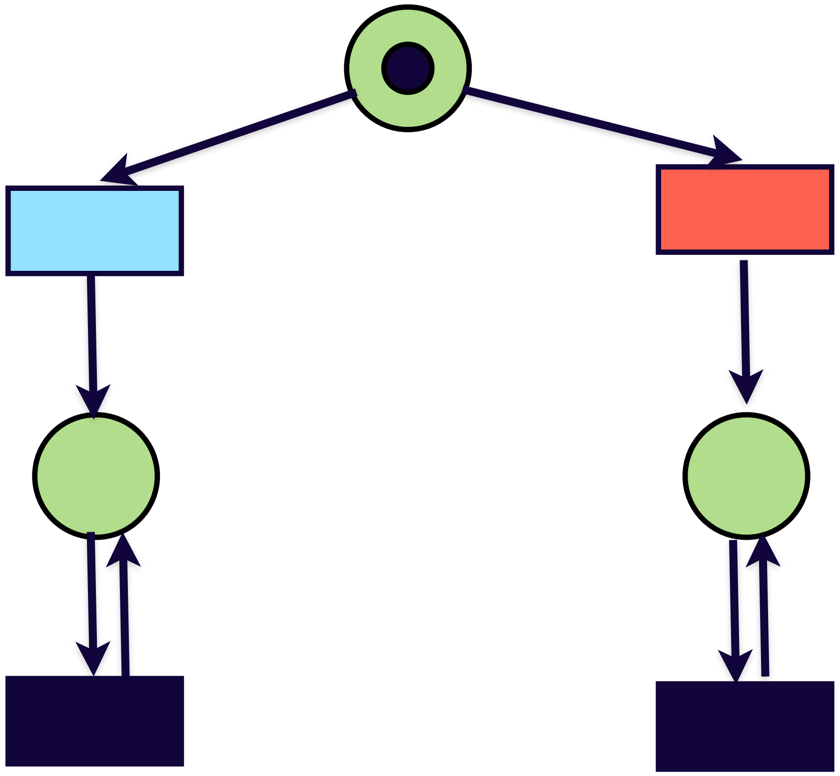
Hinweis. Sie können folgende Eigenschaft verwenden:

Sei $\bar{m}$ ein Knoten im Überdeckungsgraph, der $\bar{m}(p) = \omega$ für mehrere Plätze p erfüllt, dann ist im P/T Netz stets eine Markierung m erreichbar, die alle diese Plätze gleichzeitig *über jede Schranke n bringt:*

$$\forall n \in \mathbb{N} : \exists m \in R(N, \bar{m}_0) : m \leq_\omega \bar{m} \land (\forall p \in P : \bar{m}(p) = \omega \Longrightarrow m(p) \geq n)$$

Hierbei besagt $m_1 \leq_\omega m_2$, dass die beiden Markierungen in allen endlichen Markierungen gleich sind:

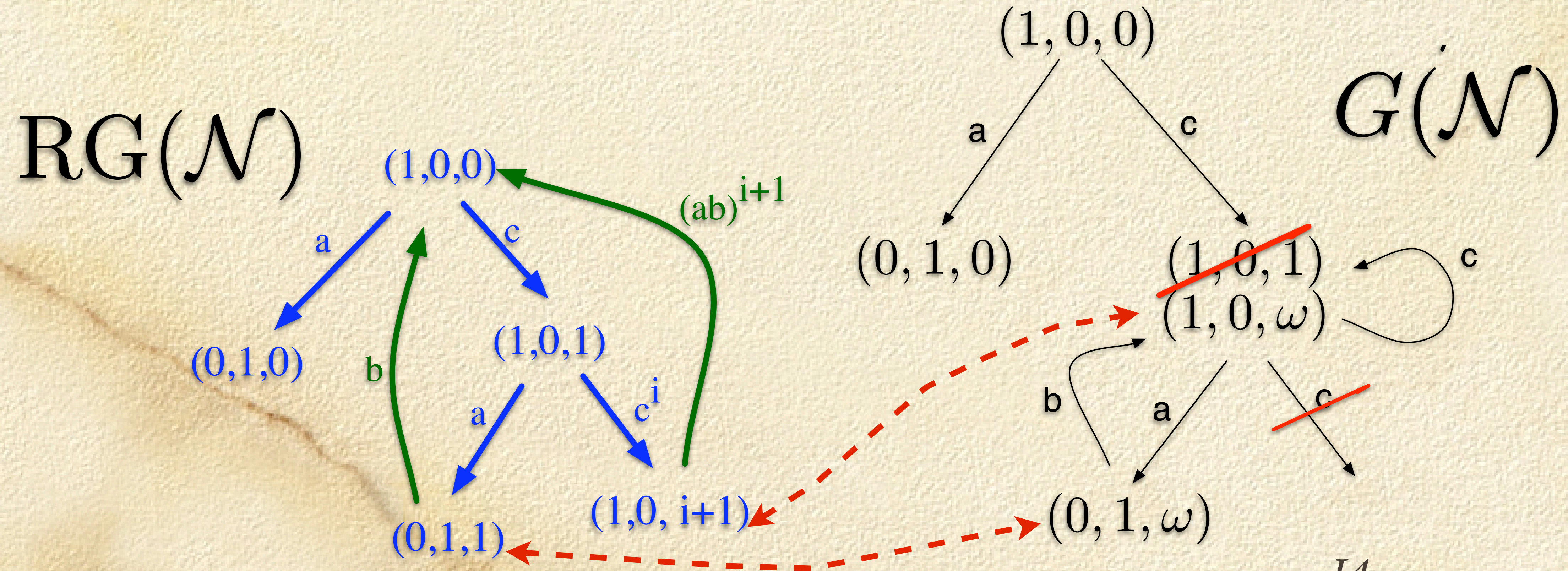
$$m_1 \leq_\omega m_2 \iff \forall p \in P : m_1(p) = m_2(p) \lor m_2(p) = \omega,$$



Die Bedeutung des Überdeckungsgraphen ergibt sich aus folgenden Eigenschaften:

Satz 3.23 *Sei $\mathcal{N} = \langle P, T, F, W, \mathbf{m}_0 \rangle$ ein P/T-Netz und $G(\mathcal{N}) = (V, E)$ ein Überdeckungsgraph zu $\mathcal{N}$, dann ist ein Platz $p \in P$ genau dann* beschränkt, *wenn es keinen Knoten $\mathbf{m} \in V$ mit $\mathbf{m}(p) = \omega$ gibt.*

> *Gilt $\mathbf{m}_0 \xrightarrow{w} \mathbf{m}$ im Netz $\mathcal{N}$ für ein Wort $w \in T^*$, so gibt es in $G(\mathcal{N})$ einen Knoten $\mathbf{m}_1$ mit $\mathbf{m}_1 \geq \mathbf{m}$ und $\mathbf{m}_0 \xrightarrow[w]{*} \mathbf{m}_1$.*



Satz 3.24 *Der Algorithmus 3.4 zur Konstruktion von* $G(\mathcal{N})$ *terminiert.*

Satz 3.26 *Für ein P/T-Netz* $\mathcal{N}$ *ist* $\mathrm{RG}(\mathcal{N})$ *genau dann endlich, wenn kein Knoten von* $G(\mathcal{N})$ *eine* ω*-Komponente besitzt.*

3.5.2 Turing-Mächtigkeit *von Petrinetzen*

Turing-Maschine

(2-)Zählerautomat

?

P/T-Netz

	Turing-mächtig?
P/T-Netze	nein
Inhibitor-Netze	ja
gefärbte Netze mit endlichen Farbmengen	nein
gefärbte Netze mit beliebigen Farbmengen	ja

Tabelle 3.2: Turing-Mächtigkeit von Petrinetzen

Definition 3.36 (Erreichbarkeitsproblem)

Gegeben: *Ein P/T-Netz* $\mathcal{N} := (P, T, F, W, \mathbf{m}_0)$ *und eine Markierung* $\mathbf{m} \in \mathbb{N}^P$.

Frage: *Gilt* $\mathbf{m} \in \mathbf{R}(\mathcal{N})$ *?*

Satz 3.39 *Das Erreichbarkeitsproblem für ~~endliche~~ beschränkte P/T-Netze ist entscheidbar,*
benötigt jedoch mindestens exponentiell viel Platz.

Eine untere Schranke $NSpace(2^{O(n)})$ wurde von Lipton 1976 bewiesen. Einen guten Überblick über weitere Komplexitätsresultate zu Algorithmen und Problemen bei Petrinetzen ist bei Esparza und Nielsen (1994) zu finden.

Definition 3.36 (Erreichbarkeitsproblem)

Gegeben: *Ein P/T-Netz* $\mathcal{N} := (S, T, W, K, \mathbf{m}_0)$ *und eine Markierung* $\mathbf{m} \in \mathbb{N}^P$.

Frage: *Gilt* $\mathbf{m} \in \mathrm{RG}(\mathcal{N})$ *?*

Definition 3.37 (Überdeckbarkeitsproblem)

Gegeben: *Ein P/T-Netz* $\mathcal{N} := (S, T, W, K, \mathbf{m}_0)$ *und eine Markierung* $\mathbf{m} \in \mathbb{N}^P$.

Frage: *Gibt es* $\mathbf{m}' \in \mathrm{RG}(\mathcal{N})$ *mit* $\mathbf{m}' \geq \mathbf{m}$ *?*