Analyse von Petri Netzen:
Erreichbarkeit, Invarianten und Simulation

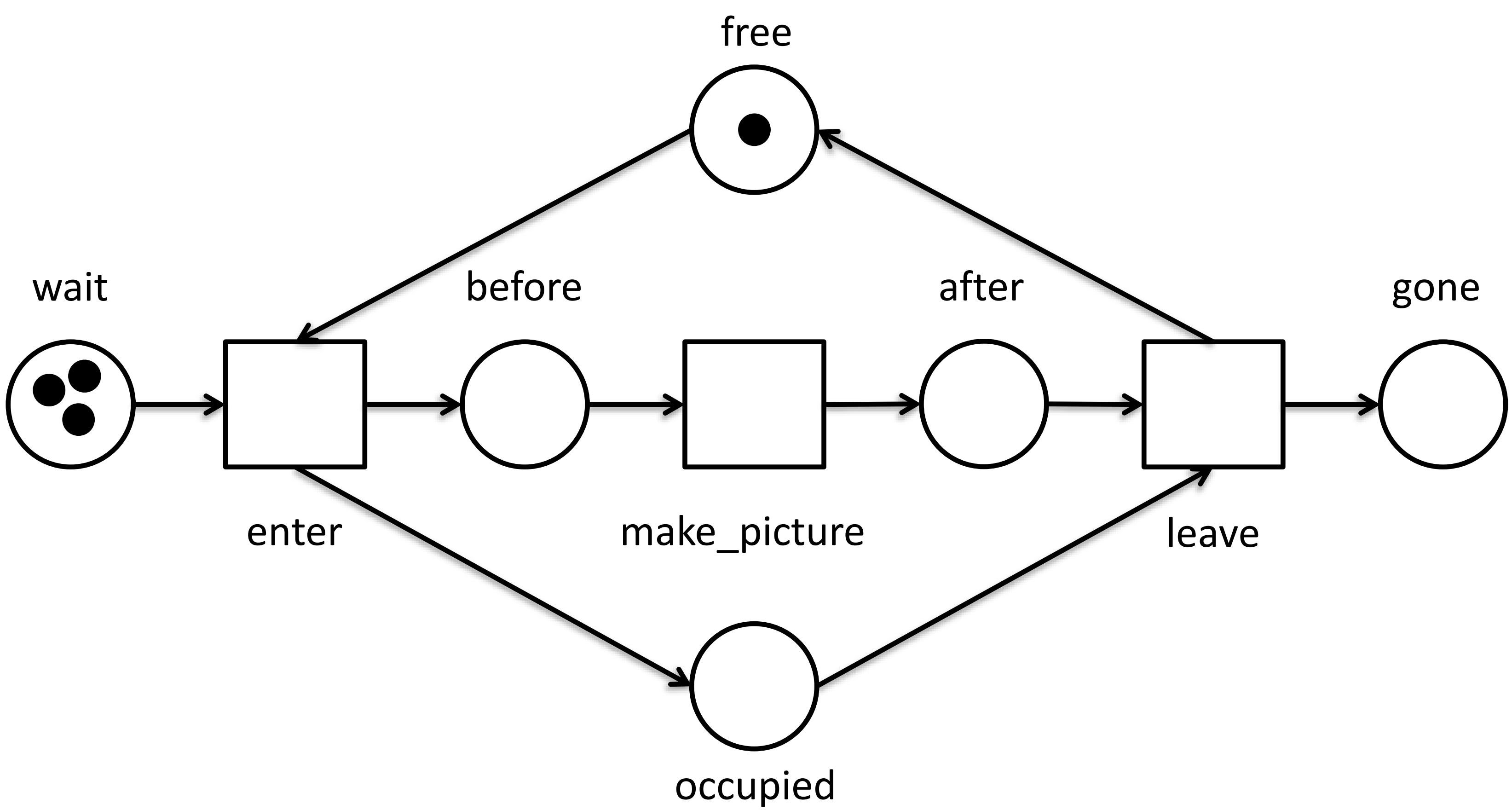
Basiert auf Material von

Prof.dr.ir. Wil van der Aalst

Eindhoven University of Technology

UNIVERSITÄT MANNHEIM

Erinnerung: Petrinetz



Erinnerung: Ausführungssemantik

Der Zustand (marking) eines Petri-Netzes ist eine Funktion s: -> I**N**

Eine Transition *t* ist im Zustand s enabled genau dann, wenn:

$$\boxed{\forall p \in P: s(p) \geq I(p,t)}$$

Die Transition t kann feuern und der daraus resultierende Zustand ist wie folgt definiert:

$$\boxed{\forall p \in P: s'(p) = s(p) - I(p,t) + O(t,p)}$$

Eine Folge von Zustandsübergängen heisst *run*

Zustände und Erreichbarkeit

Typen von Zuständen

- **Startzustand**: Initiale Verteilung der Token.
- **Erreichbarer Zustand**: Zustand, der vom Startzustand aus erreichbar ist.
- **Deadlock Zustand**: Zustand, in dem keine Transition enabled ist
- **Home Zustand**: Zustand, zu dem immer zurückgekehrt werden kann (= der von jedem Zustand aus erreichbar ist.

Wie kann man diese Zustände erkennen?

Erreichbarkeitsgraphen

- Graph, der die Erreichbarkeit von Zuständen darstellt.
- Erstellung durch Generierung erreichbarer Zustände vom Startzustand aus
- Jeder Pfad im Erreichbarkeitsgraphen entspricht einem *run*

Ereichbarkeit

- Der Erreichbarkeitsgraph G = (N,E) eines Petri-Netzes ist wie folgt definiert:
  - G = S
  - E = {(s,s') |$\exists$t$\in$T: t ist in s enabled und s' ist der Zustand, der durch das feuern von t in s entsteht}

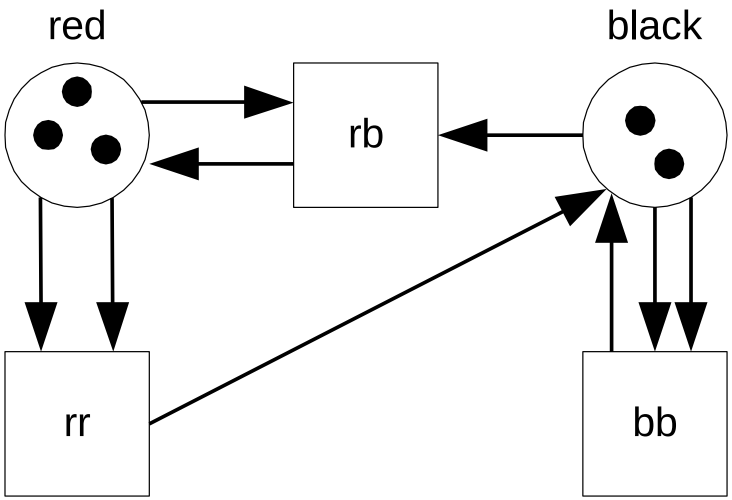
UNIVERSITÄT MANNHEIM

Konstruktion des Graphen

- Der Erreichbarkeitsgraph kann wie folgt aus einem Petrinetz erzeugt werden:
  1. Sei X die Menge, die nur den Initialen Zustand des Petri-Netzes enthält sowie Y die leere Menge
  2. Nimm ein Element x aus X und füge diesen zu Y hinzu. Berechen alle Zustände, die sich durch das feuern einer Transitionen ergibt, die enabled ist. Füge alle Zustände, die nicht in Y sind zu X hinzu.
  3. Ist X leer stoppe, sonst gehe zu 2.

UNIVERSITÄT MANNHEIM

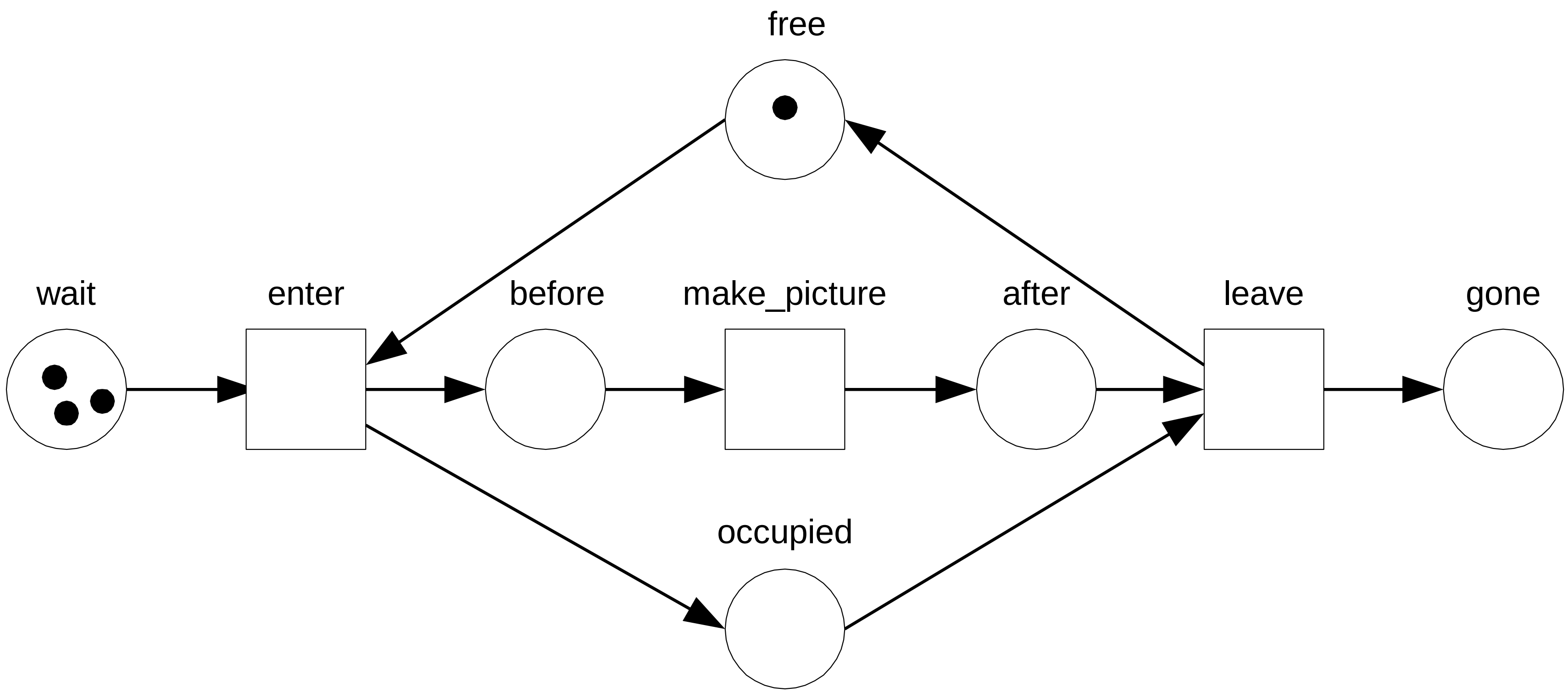
Beispiel



$$\begin{tikzcd}
(3,2) \arrow[r, "{rb,bb}"] \arrow[d, "rr"'] & (3,1) \arrow[r, "rb"] \arrow[d, "rr"'] & (3,0) \arrow[d, "rr"'] \\
(1,3) \arrow[r, "{bb,rb}"] & (1,2) \arrow[r, "{bb,rb}"] & (1,1) \arrow[d, "rb"] \\
 & & (1,0)
\end{tikzcd}$$

Knoten im Erreichbarkeitsgraphen können durch Vektoren oder durch Ausdrücke der Form „3 red + 2 black“ dargestellt werden

Übung: Konstruiere den Erreichbarkeitsgraphen



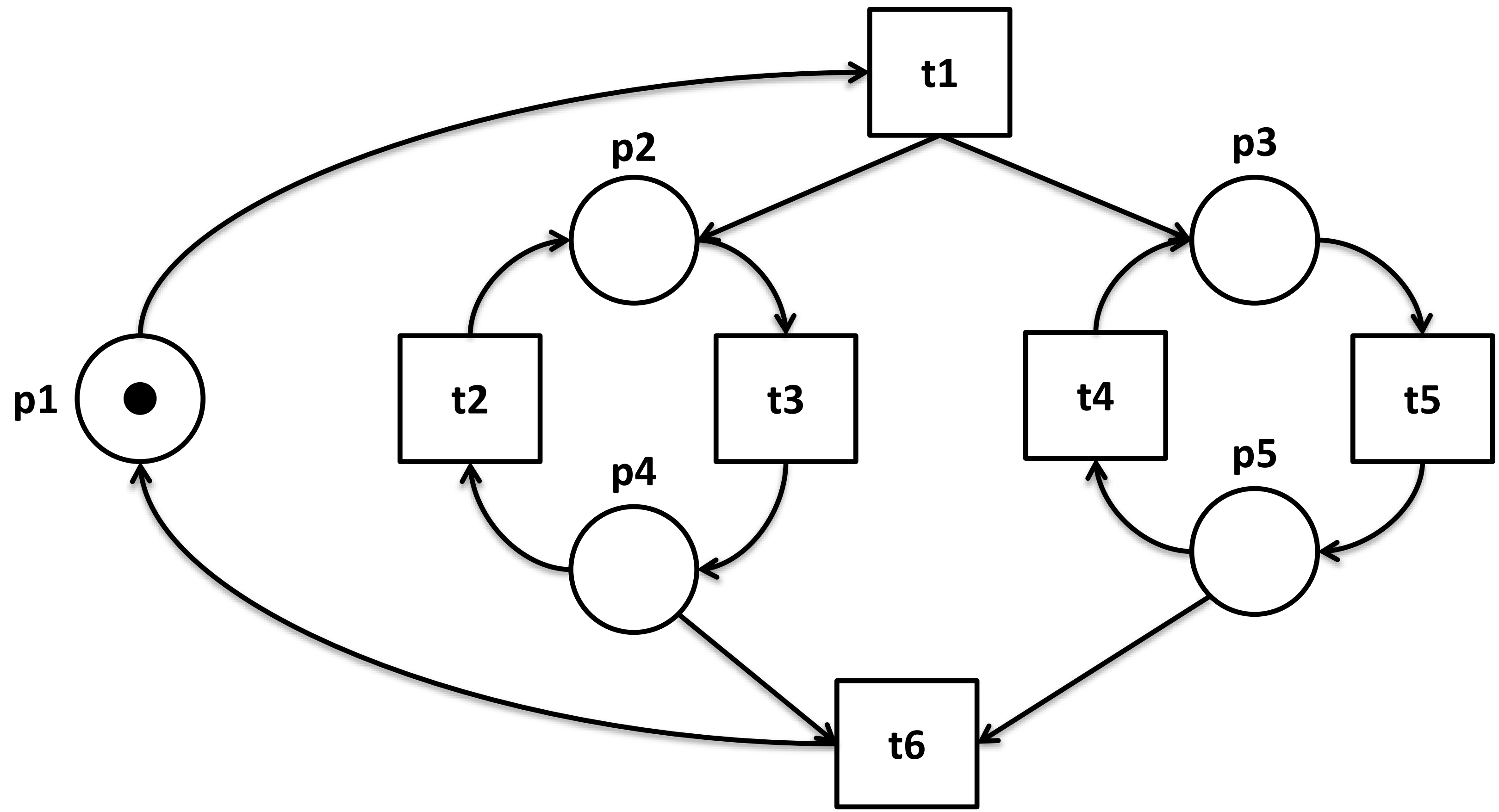
UNIVERSITÄT MANNHEIM

Netzeigenschaften

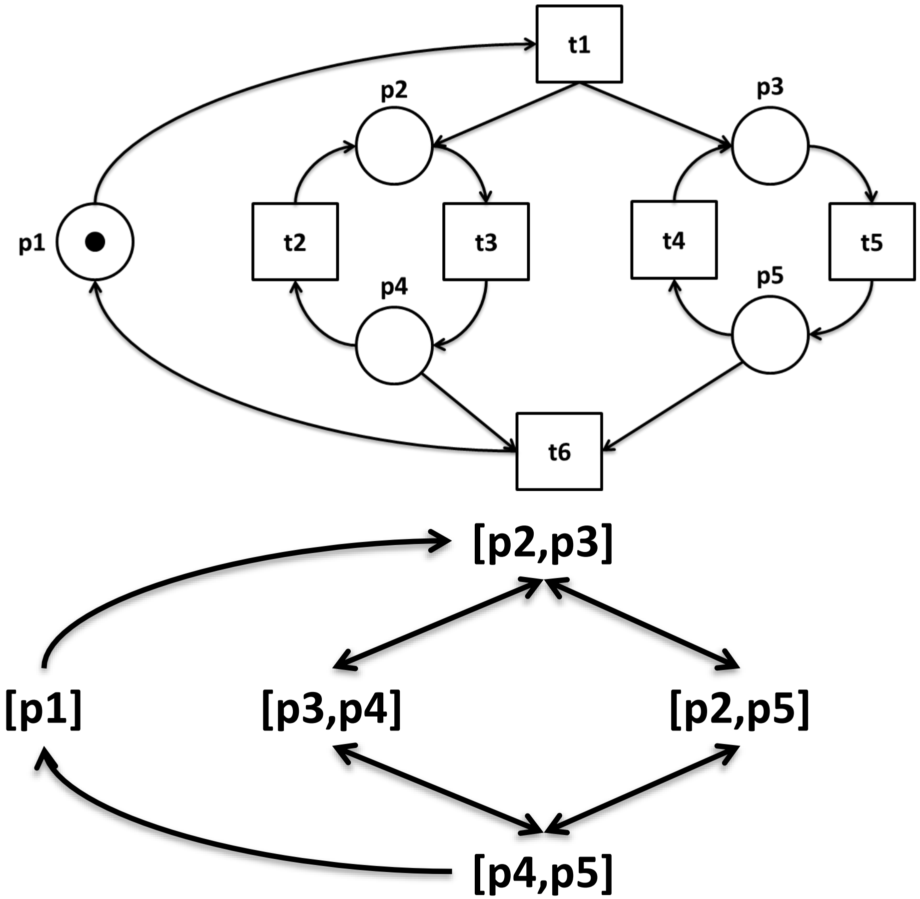
Eigenschaften des gesamten Netzes

- **Boundedness:** Ein Petrinetz ist k-bounded, wenn keine Stelle in einer erreichbaren Markierung mehr als k Marken enthält
- **Termination:** Ein Petrinetz ist terminierend, wenn jeder mögliche *run* endlich ist.
- **Deadlockfreiheit:** Ein Petrinetz ist deadlockfrei, wenn es keine Deadlockzustände enthält.
- **Reversibility:** Ein Petrinetz ist reversible, wenn der Startzustand aus jedem erreichbaren Zustand erreicht werden kann.

Beispiel



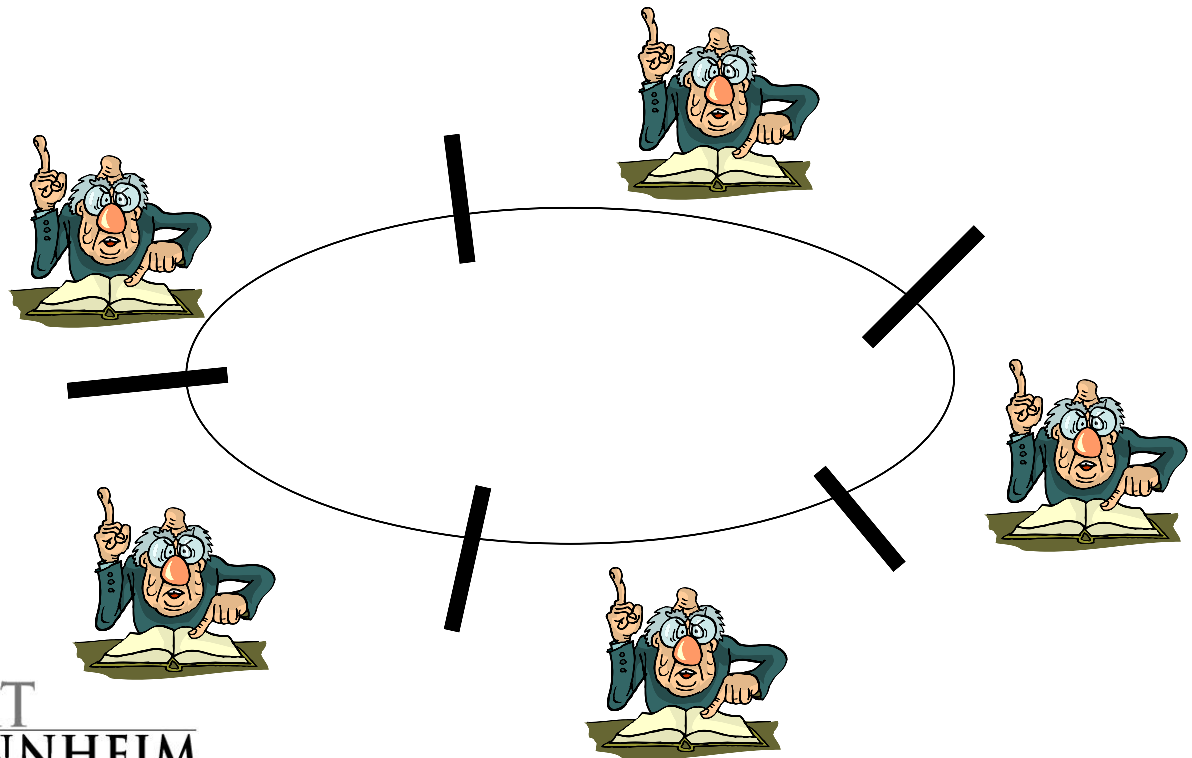
Erreichbarkeitsgraph



- ✓ Bounded?
- ✗ Terminating?
- ✓ Deadlock-free?
- ✓ Reversible?

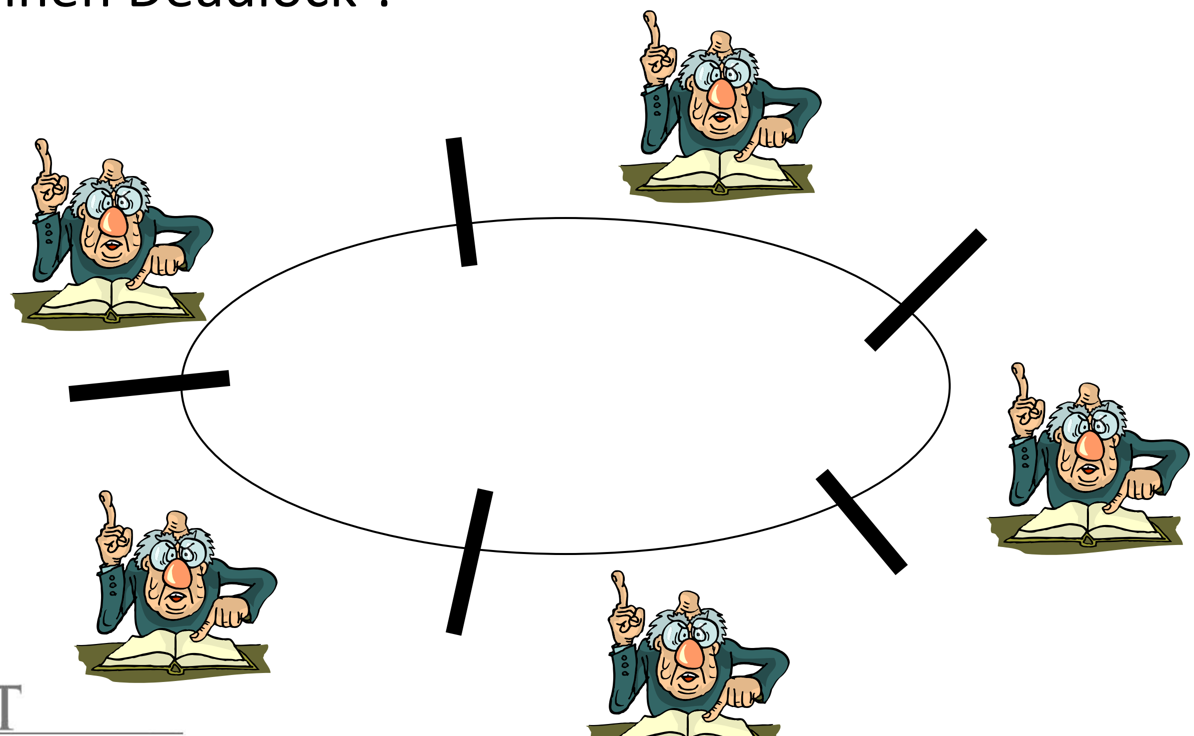
Dining Philosophers

- 5 Philosphen teilen sich 5 Stäbchen
- Ein Philosoph denkt oder Isst. Zum Essen benötigt er 2 Stäbchen



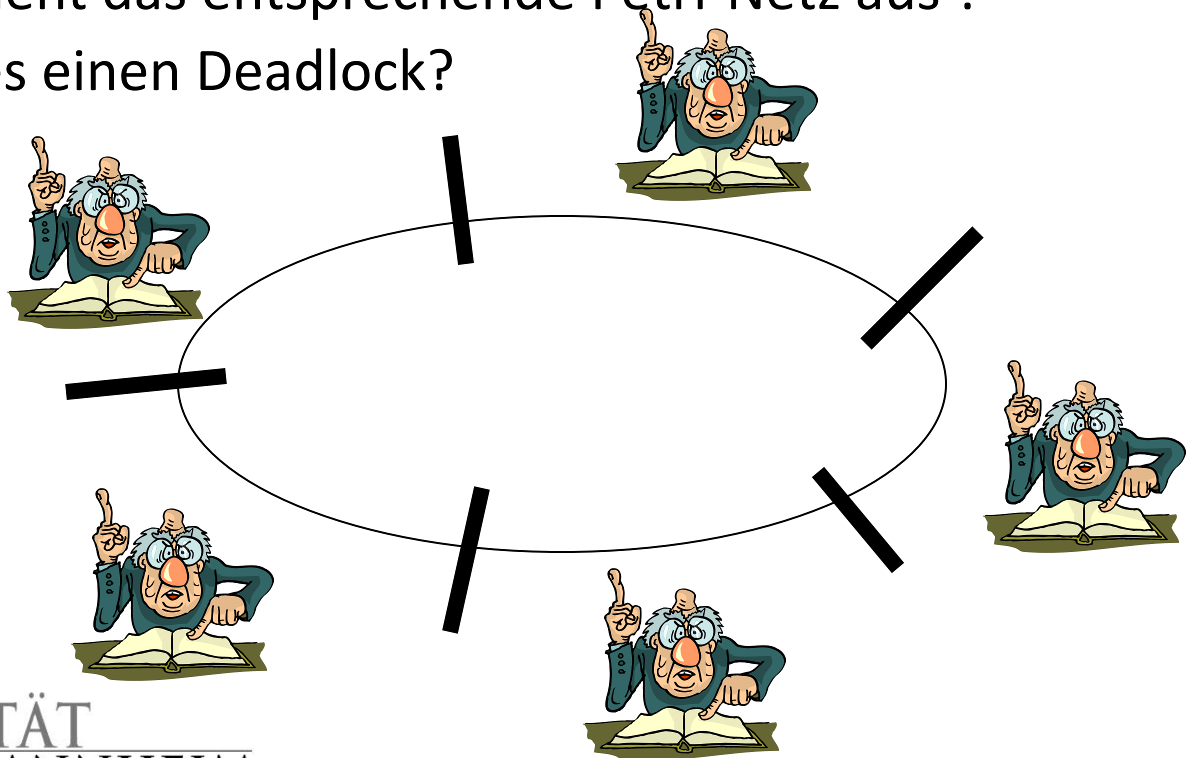
Übung: Dining philosopher

- Angenommen, Philosophen nehmen immer erst das rechte und dann das linke Stäbchen
- Wie sieht das entsprechende Petri-Netz aus ?
- Gibt es einen Deadlock ?



Übung: Dining philosopher

- Angenommen Philosophen nehmen die Stäbchen in beliebiger Reihenfolge und können diese auch wieder zurücklegen
- Wie sieht das entsprechende Petri-Netz aus ?
- Gibt es einen Deadlock?



Dead Transitions und Liveness

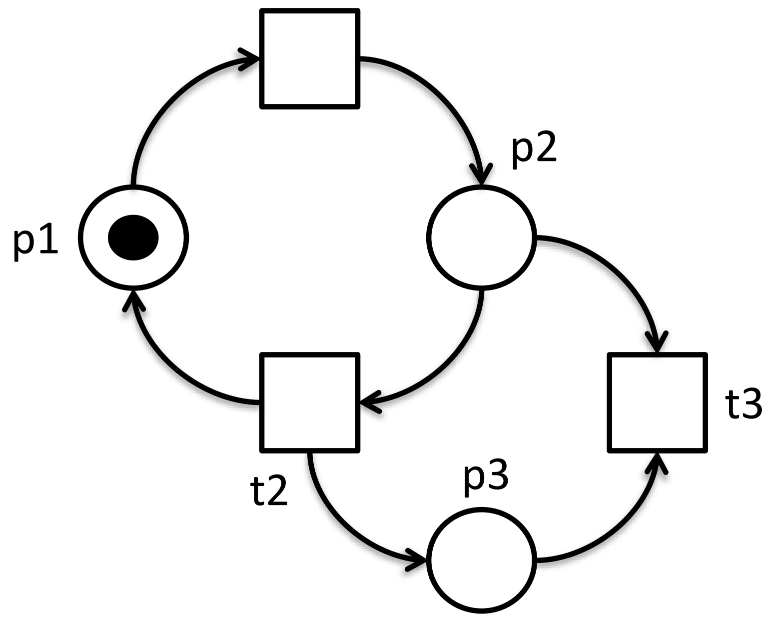
- **Dead Transistion:** Eine Transition ist dead, wenn sie in keinem erreichbaren Zustand enabled ist.
- **Live Transistion:** Eine Transition ist live, wenn es von jedem erreichbaren Zustand aus möglich ist, einen Zustand zu erreichen, in dem die Transition enabled ist.
- **Liveness:** Ein Netz ist live, wenn alle seine Transitionen live sind.

Beispiel

Dead Transitions:

Live Transitions:

→ Das Netz ist nicht Live



Unendliche Zustandsräume

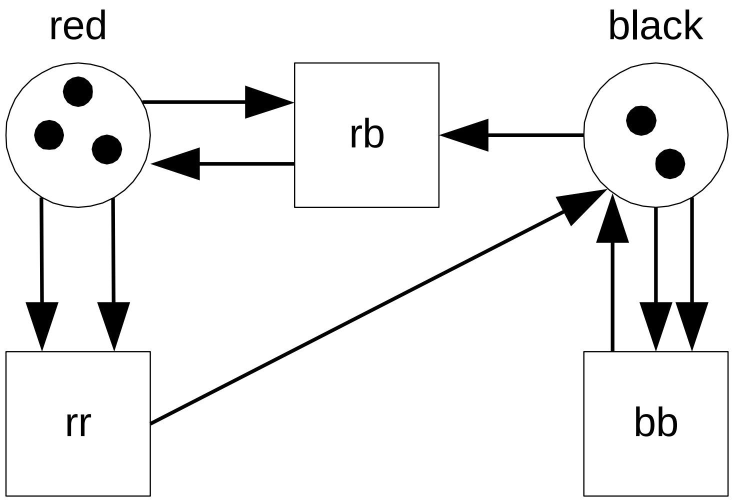
Erinnerung: Erreichbarkeitsgraphen

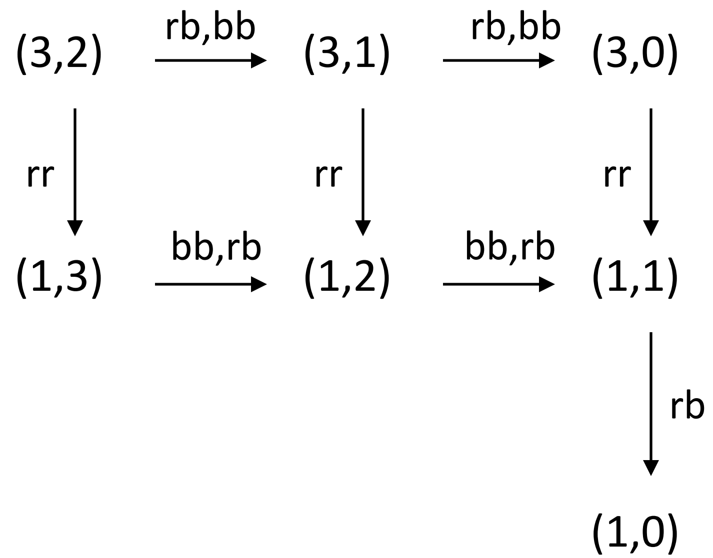
- Graph, der die Erreichbarkeit von Zuständen darstellt.

- Erstellung durch Generierung erreichbarer Zustände vom Startzustand aus

- Jeder Pfad im Erreichbarkeitsgraphen entspricht einem *run*

Erinnerung: Erreichbarkeitsgraphen

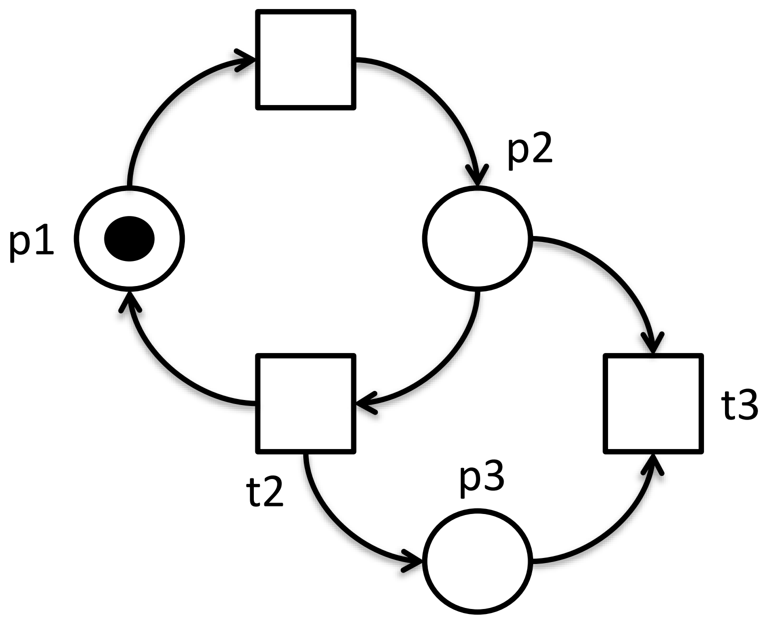




Knoten im Erreichbarkeitsgraphen können durch Vektoren oder durch Ausdrücke der Form „3 red + 2 black“ dargestellt werden

Frage

- Wie sieht der Erreichbarkeitsgraph für dieses Netz aus?



[p1]
t1 ↓
[p2]
t2 ↓
[p1,p3]
t1 ↓
[p2,p3] $\xrightarrow{t3}$ []
t2 ↓
[p1,2p3]
t1 ↓
[p2,2p3] $\xrightarrow{t3}$ [p3]
t2 ↓
[p1,3p3]
↓

Zustände für unbounded-Netze

- Erweiterung des Konzept einer Markierung:

$$m: P \rightarrow N \cup \omega$$

$$\omega + n = \omega - n = \omega, n \in N$$

Stellen können eine bestimmte, oder eine unbestimmte Anzahl von Token enthalten.

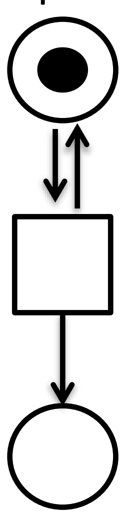
- Abdeckungsgraphen: Erreichbarkeitsgraph für das erweiterte Konzept der Markierung

- Wie erkennen wir, dass ein Zustand unbounded ist ?

Abdeckungsgraphen

- Analyse von *unbounded* Petri Netzen

Petri-Netz:



Erreichbarkeitsgraph:
(unendlich)

[p1]
↓
[p1,p2]
↓
[p1, 2p2]
↓
[p1, 3p2]
↓
…

Abdeckungsgraph:
(endlich)

[p1]
↓
[p1,ωp2]

$[p1] \leq [p1,p2]$
$0 \cdot p2 < 1 \cdot p2$
→ [p1, ωp2]

Konstruktion von Abdeckungsgraphen

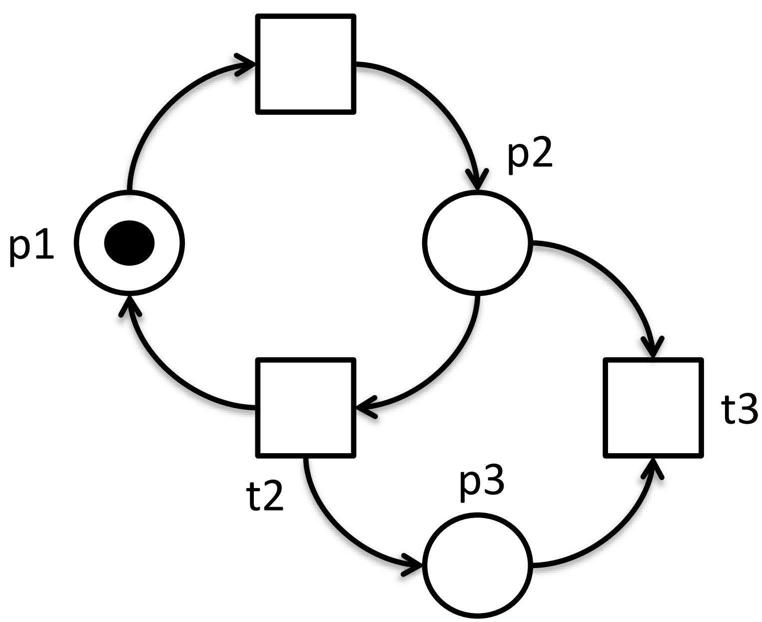
- Der Erreichbarkeitsgraph kann wie folgt aus einem Petrinetz erzeugt werden:

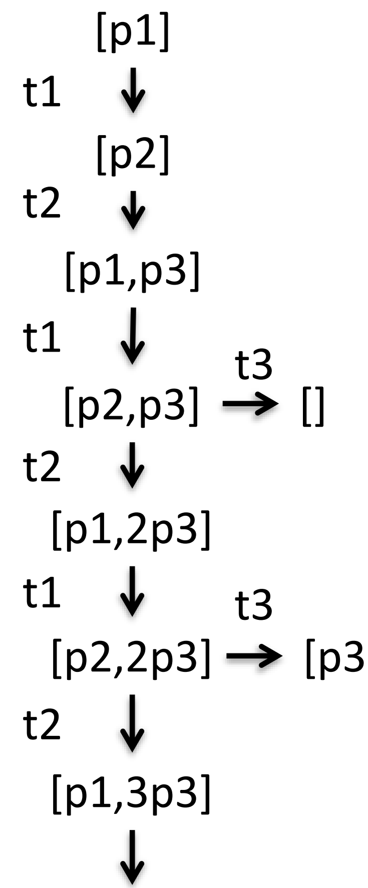
  1. Sei X die Menge, die nur den Initialen Zustand des Petri-Netzes enthält sowie Y die leere Menge
  2. Nimm ein Element x aus X und füge diesen zu Y hinzu.
  3. Berechen alle Zustände, die sich durch das feuern Transitionen ergibt, die enabled sind.
  4. Für jede Markierung m, ersetze die Anzahl der Tokens der Stelle p durch ω falls es eine Markierung m‘ ≤ m gibt, in der m‘(p) > m(p) .
  5. Füge alle Zustände, die nicht in Y sind zu X hinzu.

  Ist X leer stoppe, sonst gehe zu 2.

Aufgabe

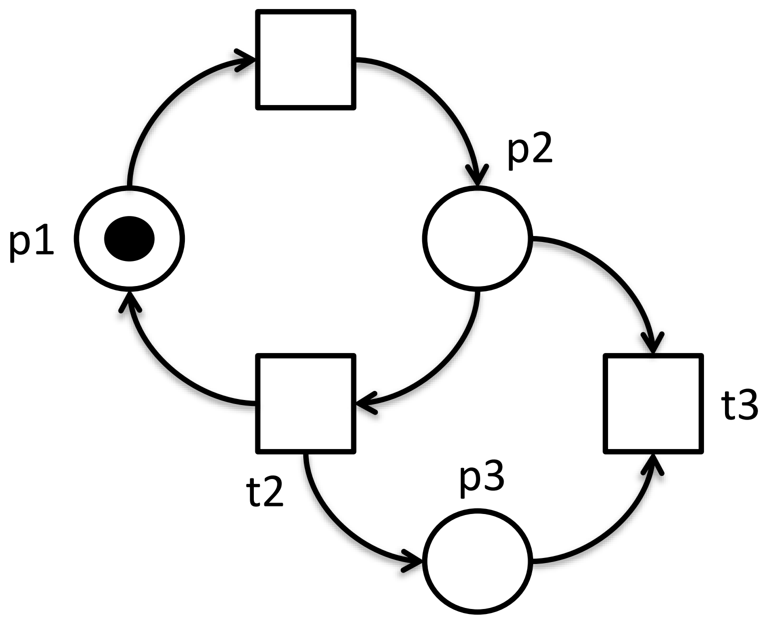
- Wie sieht der Abdeckungsgraph für dieses Netz aus?

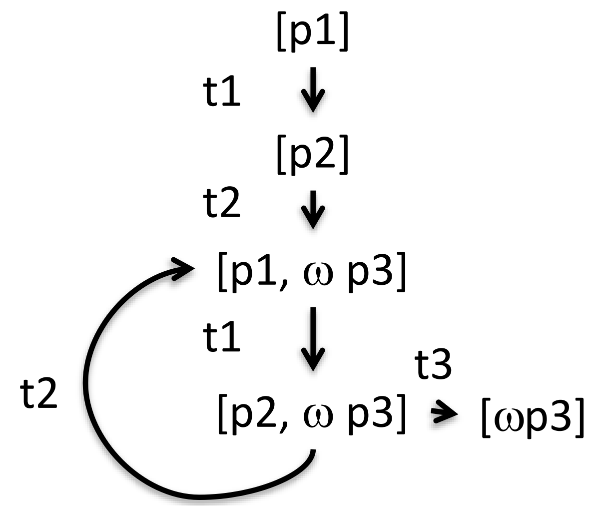




Aufgabe

- Wie sieht der Abdeckungsgraph für dieses Netz aus?





Eigenschaften des Abdeckungsgraphen

- Ereichbarkeits- und Adeckungsgraph eines *bounded* Petri Netzes sind identisch

- Der Abdeckungsgraph ist immer endlich

- Eine Transistion t ist *dead* genau dann, wenn sie nicht im Abdeckungsgraphen auftritt

- Eine Stelle ist k-bounded genau dann wenn sie in keiner Markierung im Abdeckungsgraphen mehr als k Token enthält

Fairness

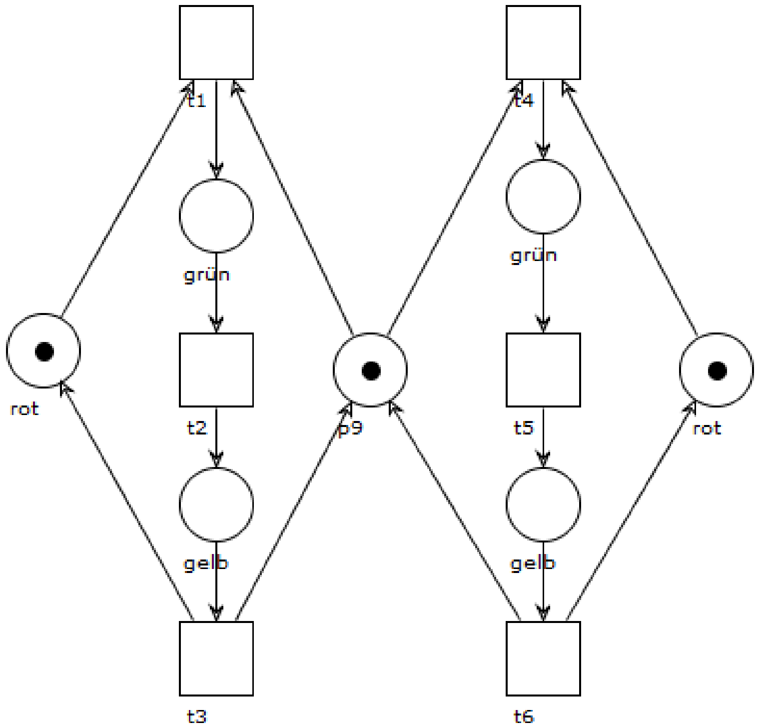
Fairness von Prozessen

- Dining Philosophers: unfair – Philosoph kann verhungern

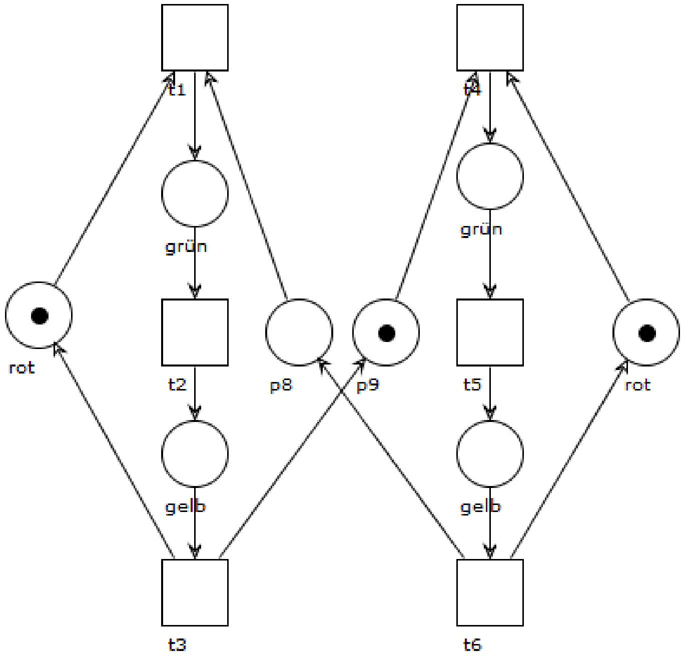
- Eigenschaften von Transitionen in unendlichen runs:
  - **Impartial:** wird in jedem unendlichen run unendlich oft ausgeführt
  - **Fair:** wird in jedem unendlich run, in dem die Transition unendlich oft *enabled* ist unendlich oft ausgeführt

Fair vs. Unfair

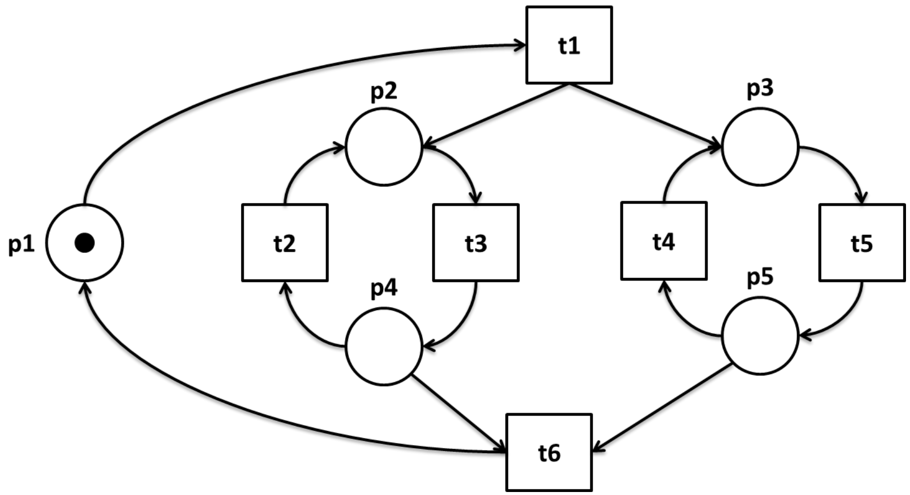
Unfair:

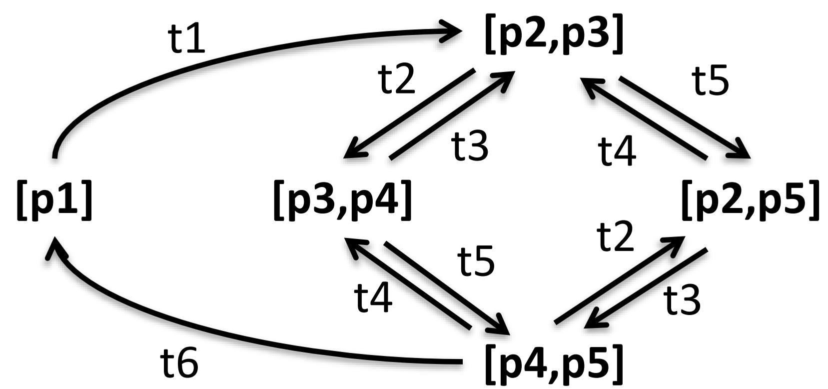


Fair:



Beispiel

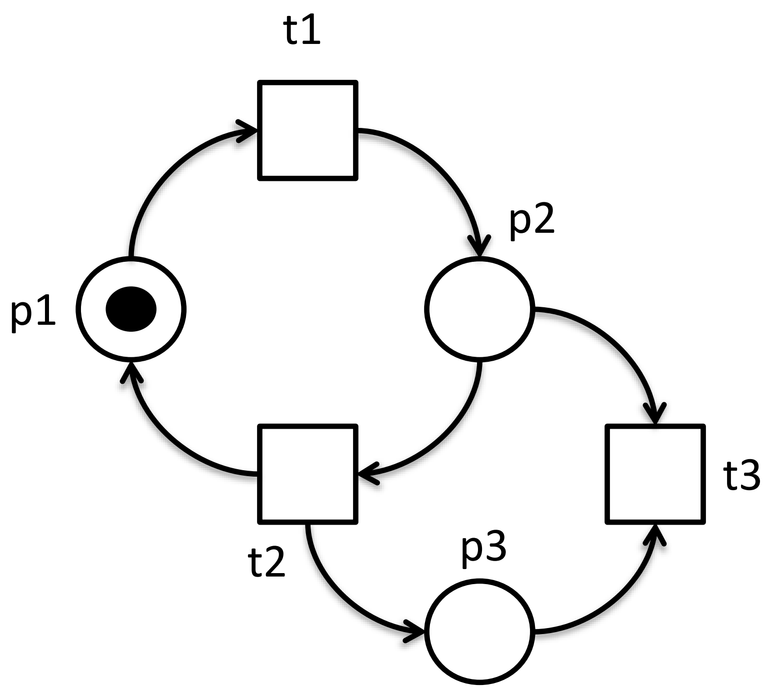


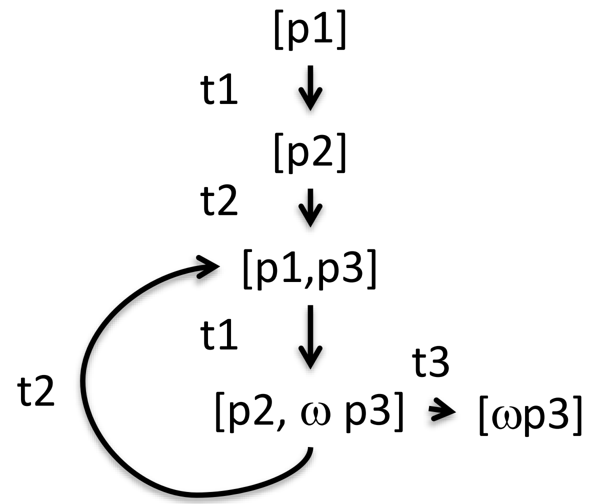


t1: fair

T2-t6: not fair

Beispiel:

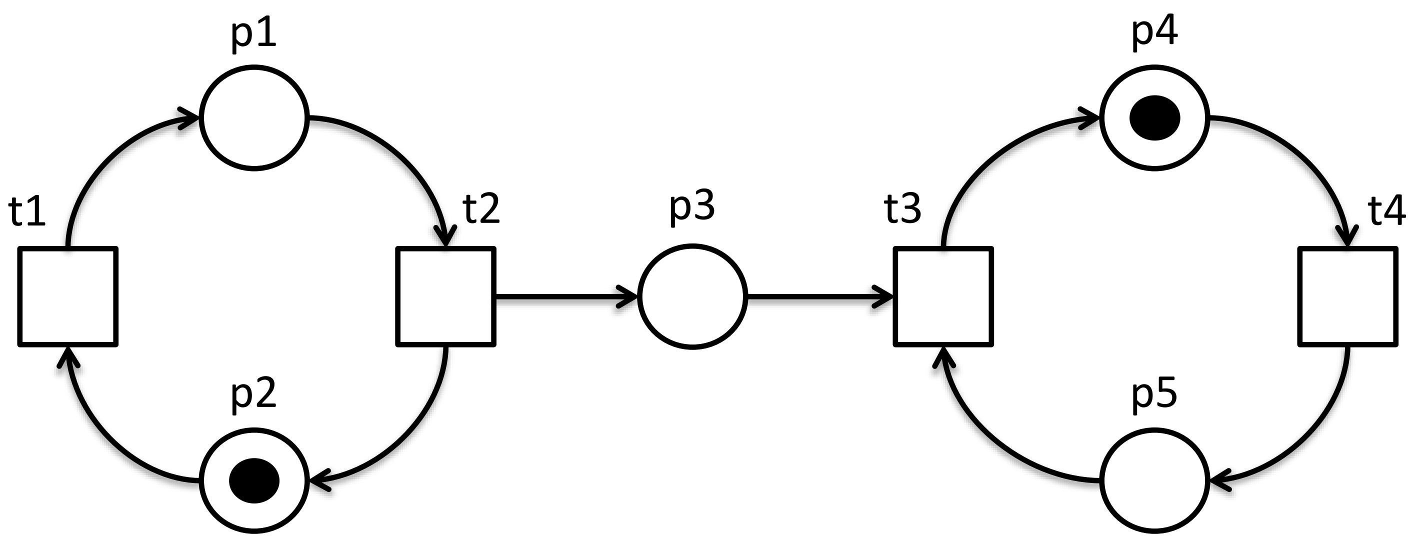




t1: impartial
t2: impartial
t3: not fair

Übung

Welche Transitionen sind *impartial*, bzw. *fair*?



Vektordarstellung und Invarianten

Petri Netze und Vektoraddition

Vektordarstellung von Markierungen:

$$\vec{m} = \begin{pmatrix} u_1 \\ \vdots \\ u_k \end{pmatrix}, u_i = m(p_i)$$

Vektordarstellung von Transitionen:

$$\vec{t} = \begin{pmatrix} v_1 \\ \vdots \\ v_k \end{pmatrix}, v_i = O(t, p_i) - I(p_i, t)$$

Matrixdarstellung des Netzes:

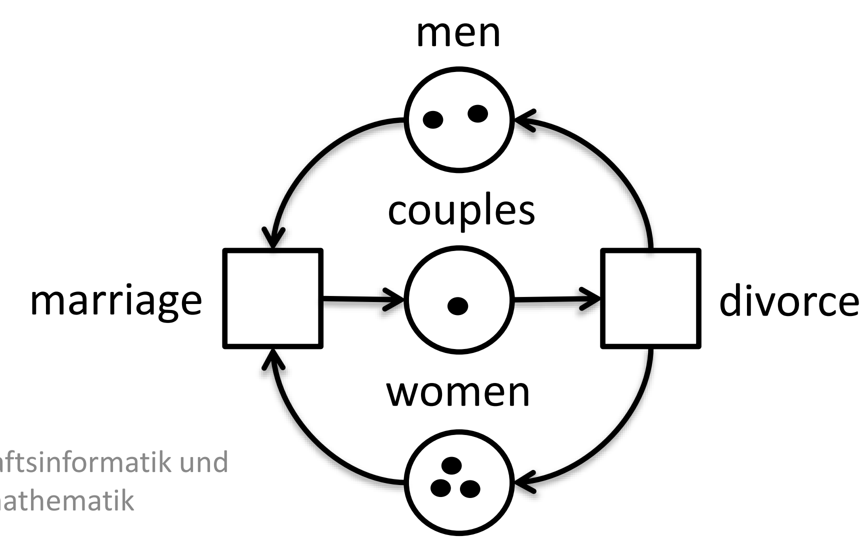
$$C = \left(\overrightarrow{marriage} \quad \overrightarrow{divorce}\right)$$

m + marriage = m‘

$$\begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix}$$

m + marriage + divorce = m

$$\begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$



Erreichbarkeit und Invarianten

- Durch eine bestimmte Kombination von Transitionen erreichbarer Zustand:

$$\overrightarrow{m} = \overrightarrow{m_0} + C * \vec{x}$$

- X ist ein Vektor mit der Anzahl der Ausführungen der verschiedenen Transitionen.

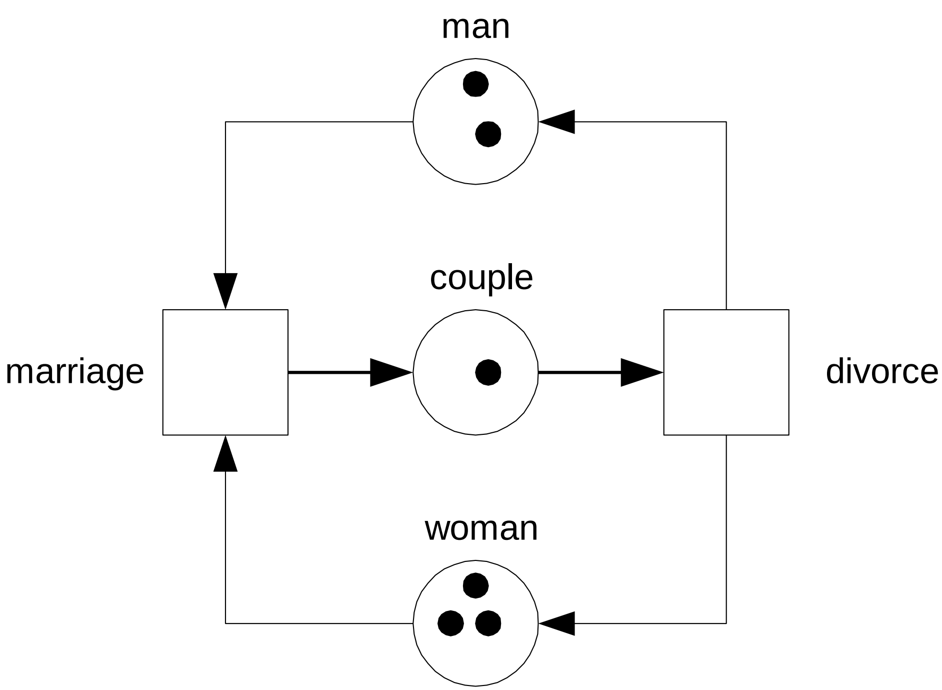
$$\begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} + \mathrm{C} * \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

- Bietet einen Ansatz zur Berechnung von Invarianten durch lineare Gleichungssysteme

Strukturelle Analyse

- Vermeidet Probleme der Zustandsanalyse, insbes. Größe der generierten Graphen.
- Erfasst Eigenschaften, die unabhängig vom Anfangszustand sind.
- Wir betrachten:
  - Stelleninvarianten
  - Transitionsinvarianten
- Invarianten können mit Hilfe von Methoden der linearen Algebra ermittelt werden

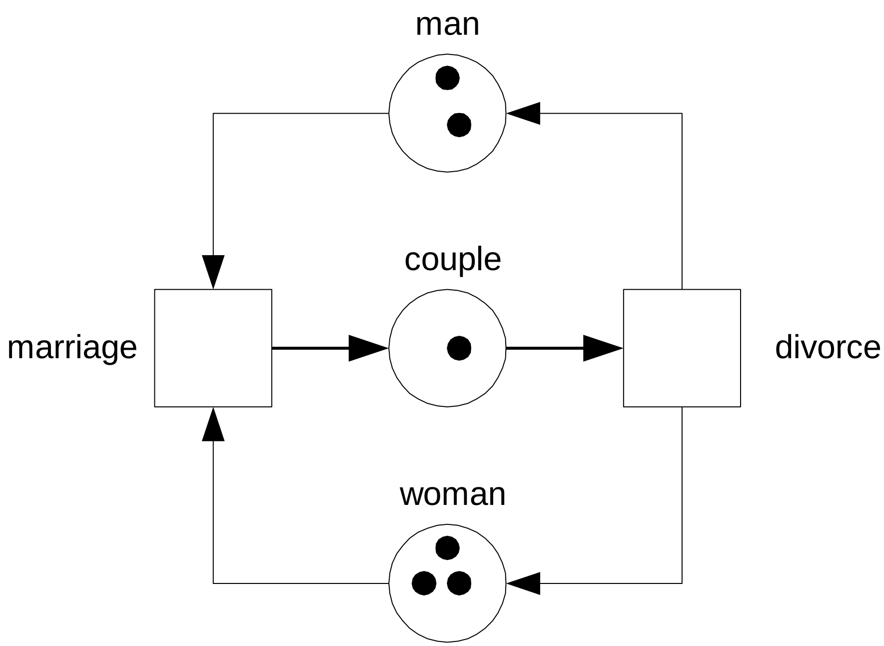
Stelleninvarianten



- Jeder Stelle wird ein Gewicht zugeordnet.
- Das Gewicht eines Tokens entspricht dem Gewicht der Stelle.
- Die *gewichtete Summe* der Token bleibt durch transitionen unverändert (invariant)

1 man + 1 woman + 2 couple

Beispiele für Invarianten



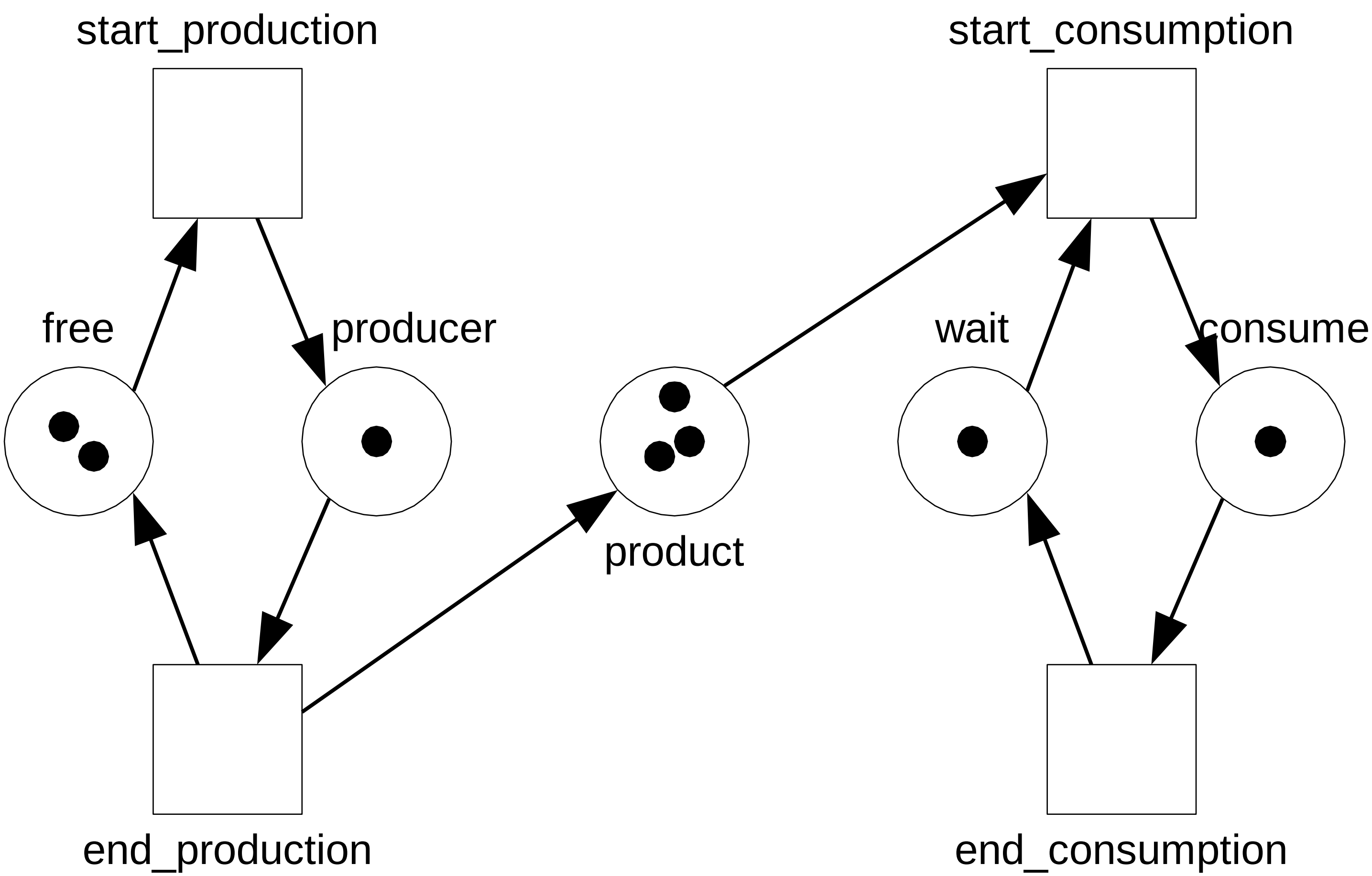
- 1 man + 0 woman + 1 couple

(Also denoted as: man + couple)

- 2 man + 3 woman + 5 couple
- -2 man + 3 woman + couple
- man – woman
- woman – man

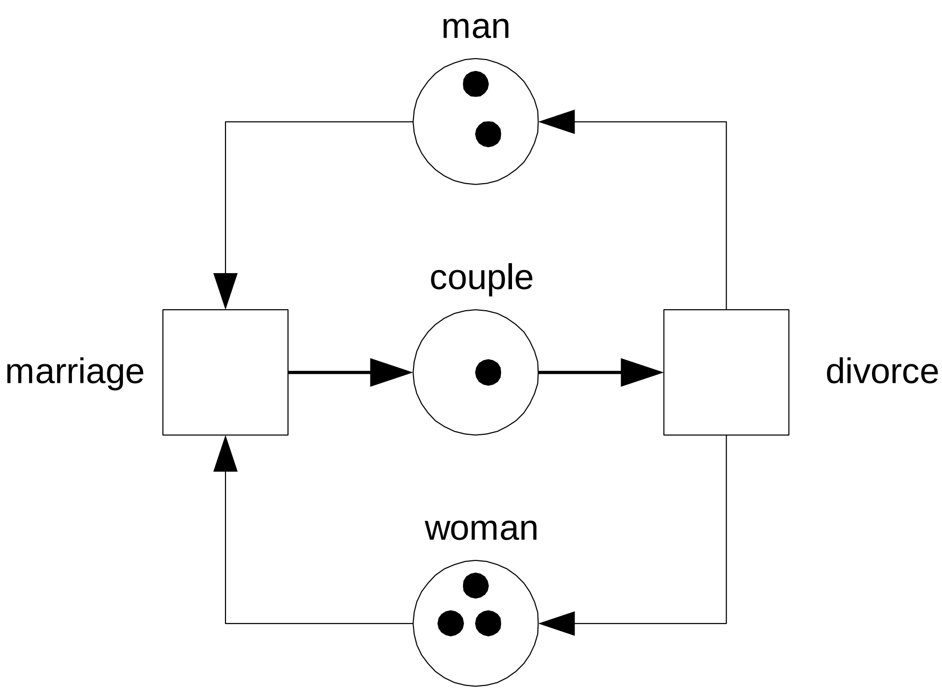
(Jede lineare Kombination von Invarianten ist eine Invariante.)

Übung: Bestimme Stelleninvarianten



UNIVERSITÄT MANNHEIM

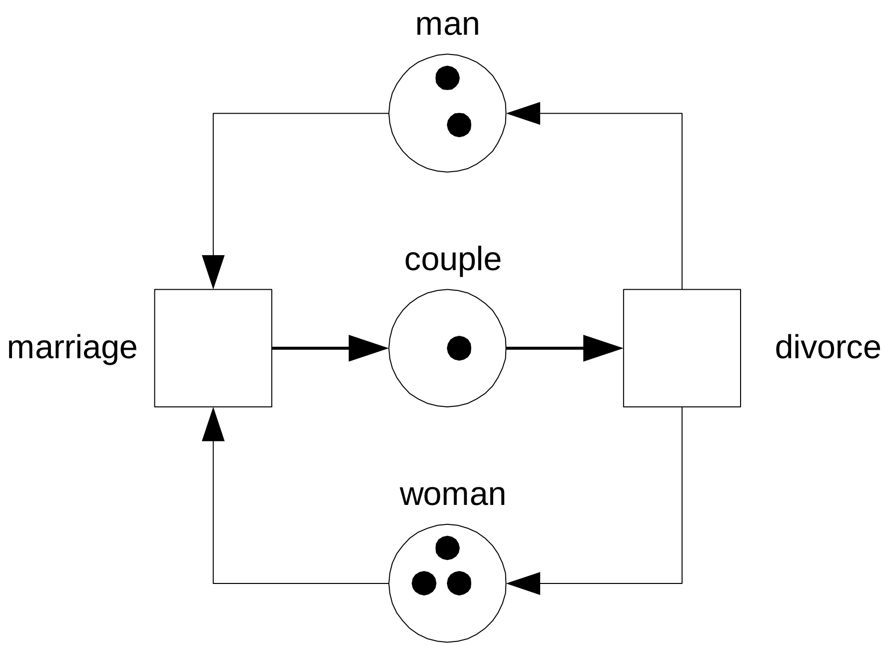
Transitionsinvarianten



- Jeder Transitionen wird ein Gewicht zugewiesen.
- Feuert jede Transition die angegebene Anzahl von malen, ist das System wieder im Ausgangszustand.
- D.h. diese Invarianten beschreiben mögliche Transistionsmengen

2 marriage + 2 divorce

Andere Invarianten



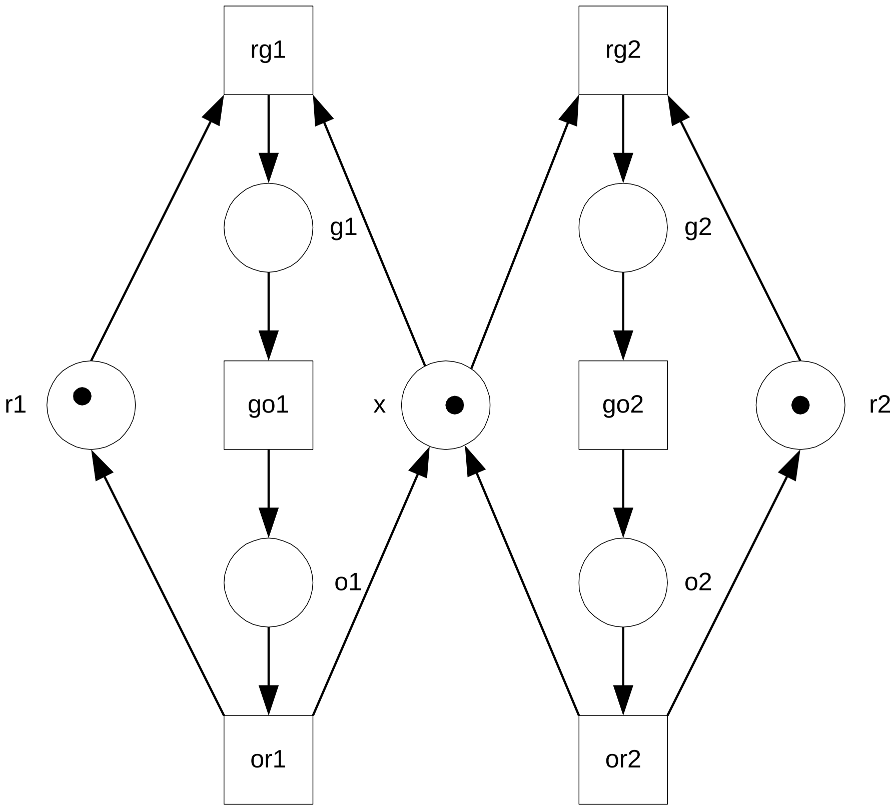
- 1 marriage + 1 divorce

(Also denoted as: marriage + divorce)

- 20 marriage + 20 divorce
- Bemerkungen
  - Jede lineare Kombination von Transitionsinvarianten ist wieder eine Transitionsinvariante.
  - Negative Gewichte haben jedoch keine sinnvolle Interpretation
  - Invarianten können nicht immer tatsächlich ausgeführt werden!

UNIVERSITÄT MANNHEIM

Beispiel: Ampelschaltung



Analyse mit Invarianten

- Invarianten können verwendet werden, um indirekt Eigenschaften von Netzen zu überprüfen:
  - Eine Stelle p ist *bounded* wenn es eine semipositive Stelleninvariante gibt, in dem p ein positives Gewicht hat
  - Wenn es eine positive Stelleninvariante gibt, dann ist das Netz für jede mögliche Startmarkierung *bounded*
  - Ist ein Netz *live* und *bounded*, dann hat es eine positive Transitionsinvariante, die alle Transitionen beinhaltet

Bestimmung von Invarianten

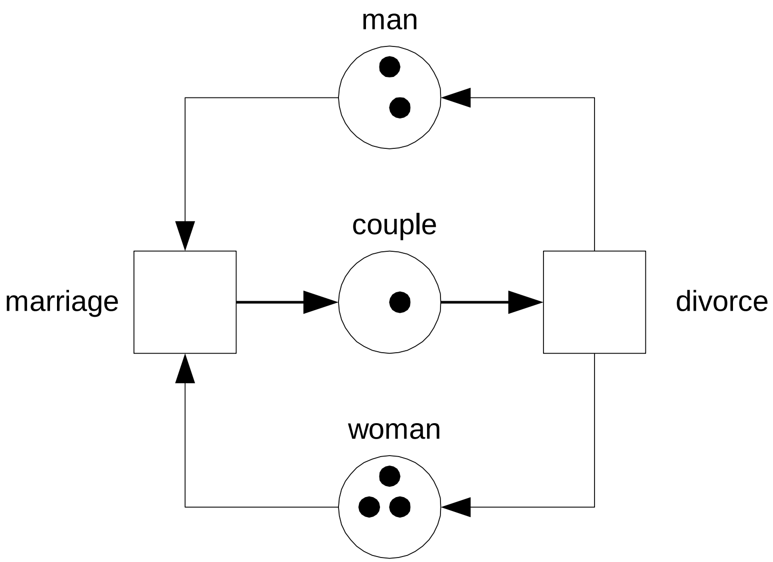
Bestimmung von Invarianten

- “intuitive Bestimmung": Formuliere eine Hypothese und überprüfe diese.
- “algebraische Bestimmung": Bestimmung durch Lösung eines linearen Gleichungssystems

Menschen bevorzugen den intuitiven Weg, Computer den algebraischen

Inzidenzmatrizen

- Reihen → Stellen.
- Spalten → Transitionen.
- Wert ij in der Matrix geben die Veränderung der Tokenmenge in der Stelle i bei feuern der Transition j an.



$$N = \begin{pmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{pmatrix}$$

UNIVERSITÄT MANNHEIM

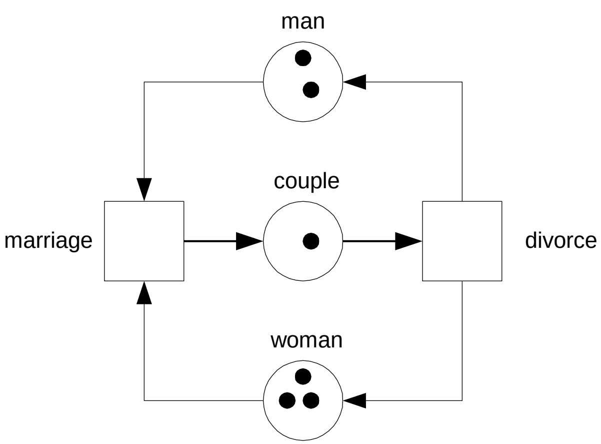
Stelleninvarianten

- Sei N eine Inzidenzmatrix mit n Stellen und m Transitionen

- Jede Lösung der Gleichung X*N = 0 ist eine Stelleninvariante
  - *X* ist ein Stellenvektor (i.e., *1 x n* Matrix)
  - *O* ist ein Transitionsvektor (i.e., *1 x m* Matrix)

- Berechnung der Basis (polynomiell)

Beispiel



$$X\begin{pmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} = (0,0)$$

$$(man, woman, couple)\begin{pmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} = (0,0)$$

Lösungen:

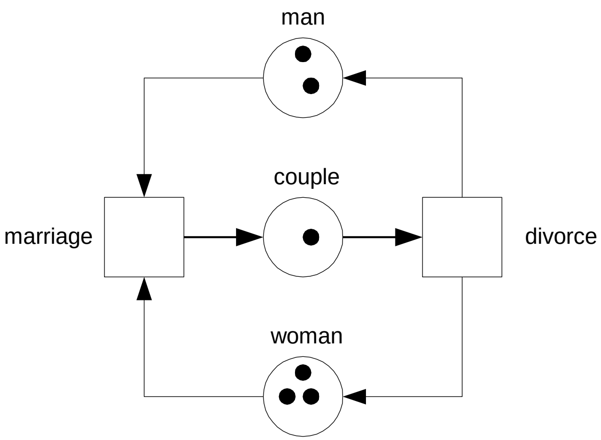
- (0,0,0)
- (1,0,1)
- (0,1,1)
- (1,1,2)
- (1,-1,0)

Transitionsinvarianten

- Sei N eine Inzidenzmatrix mit n Stellen und m Transitionen
- Jede Lösung der Gleichung N*X = 0 ist eine Transitionsinvariante
  - *X* ist ein Transitionsvektor (i.e., *m x 1* matrix)
  - *0* ist ein Stellenvektor (i.e., *n x 1* matrix)
- Berechnung der Basis (polynomial)

UNIVERSITÄT MANNHEIM

Beispiel



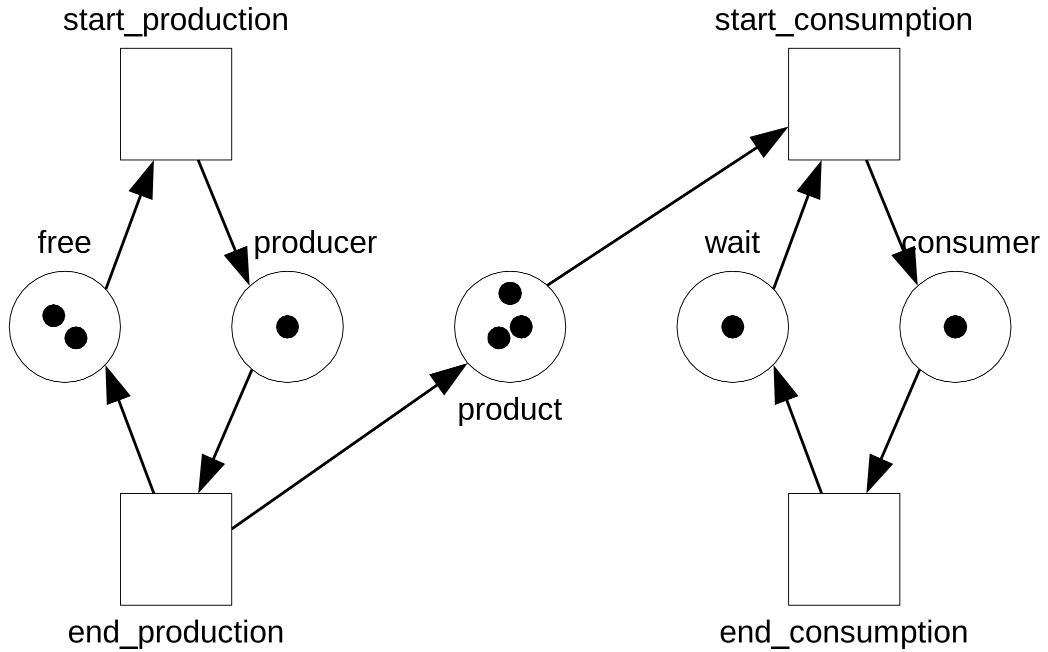
$$\begin{pmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} X = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} marriage \\ divorce \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Lösungen:

- $(0,0)^T$
- $(1,1)^T$
- $(32,32)^T$

Übung



- Gib die Inzidenzmatrix an.
- Berechne/Bestimme die Stelleninvarianten.
- Berechne/bestimme die Transitionsinvarianten.