9 Formale Modelle: Petri-Netze

9.1 Einführung

Nebenläufige Arbeit $\boxed{\text{„}\textit{Concurrency}\text{“}}$
vs. **Sequentielle Arbeit**

Endlicher Automat:

- Kausale Abfolge:
  globale Zeitskala
- Gesamtsicht:
  globaler Zustandsbegriff

Nicht adäquat für Rechnernetze, Produktionsabläufe, parallele Prozesse, Threads,...

Nebenläufigkeit

Beispiel: Berechnung Pascalsches Dreieck

$$
\begin{array}{ccccccccccccc}
 & & & & & & 1 & & & & & & \\
 & & & & & 1 & & 1 & & & & & \\
 & & & & 1 & & 2 & & 1 & & & & \\
 & & & 1 & & 3 & & 3 & & 1 & & & \\
 & & 1 & & 4 & & 6 & & 4 & & 1 & & \\
 & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & \\
1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1
\end{array}
$$

dabei

- kausal abhängige Berechnungen:
  Reihenfolge vorgeschrieben:
  "Kausalketten", z.B.
  `1+1=2` vor `1+2=3` vor `3+3=6` vor `4+6=10`
- voneinander unabhängige Berechnungen:
  Reihenfolge beliebig, evtl. auch gleichzeitig, z.B.
  `1+4=5`, `10+10=20`, `5+1=6`

Relationen:

Nebenläufigkeit :

(reflexive) symmetrische Relation

i.a. aber nicht transitiv
(“Gleichzeitigkeit” wäre transitiv)

$$nblf(a, b)$$

kausale Abhängigkeit :

irreflexiv, transitiv

i.a. nicht symmetrisch, nicht reflexiv

d.h. Abläufe darstellbar als Halbordnungen

(Spezialfall: Ordnung bei sequentiellen Abläufen)

b abhängig von a (b zeitlich nach a):

$$abh(a, b)$$

a, b unabhängig voneinander:

$$unabh(a, b) := \neg abh(a, b) \wedge \neg abh(b, a)$$

dabei gegenseitiger Ausschluß möglich,
aber Reihenfolge unbestimmt

Konflikt :

symmetrisch, nicht refelexiv, i.a. nicht transitiv

$$knfl(a, b)$$

evtl. auf Mengen beziehen
(bei mehreren Ressourcen)

Probleme:

- Parallelität, Gleichzeitigkeit
- Nebenläufigkeit
- Gegenseitiger Ausschluß
- Konflikt
- Synchronisation:
  Zusammenführen von Kausallinien
- Deadlock
- Livelock, Fairness
- Lebendigkeit

Analyse paraleller/nebenläufiger/nicht-deterministischer Programme bzw. Abläufe

zusätzliche Programmkonstruktion ($\Rightarrow$ Threads):

$$a \,||\, b \qquad a, b \text{ nebenläufig}$$

Ergebnis von

```
x=1;
( ( while (x!=0) do x = x+1;) || x = 0 ; )
```

abhängig von Fairneß-Bedingungen

Beschreibungsformen nebenläufiger Abläufe

Nicht-Determinismus

(*nondeterministic interleaving*)

"Alle möglichen Aufzeichnungen eines Beobachters"

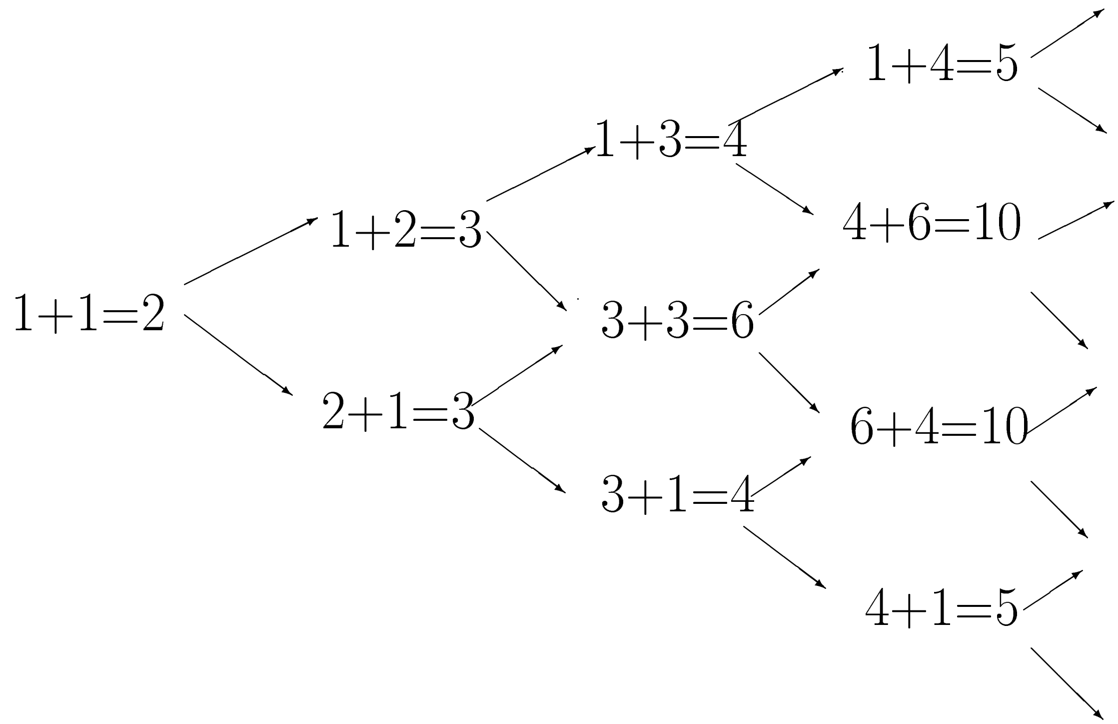
`1+1=2 1+2=3 1+3=4 1+4=5` ...
bzw. `1+1=2 1+2=3 2+1=3 1+3=4 3+1=4` ...
bzw. `1+1=2 1+2=3 1+3=4 2+1=3 3+1=4` ...
usw.

kausale Abhängigkeiten bleiben erhalten

ansonsten:
beliebige (nicht-deterministische) Reihenfolge

ergibt: Sprache L der möglichen Folgen

Beschriftete Halbordnungen



Formal komplizierter (z.B. Verkettung?)

9.2 Bedingungen und Ereignisse

Bedingungen

- gelten/gelten nicht
- mehrere Bedingungen können nebenläufig gelten
  Fall = Menge geltender Bedingungen

Ereignisse

- können nebenläufig stattfinden:
  Schritt = Menge nebenläufiger Ereignisse
- verändern Bedingungen

Zu jedem Ereignis e gibt es

Vorbedingungen : $pre(e)$
müssen gelten, damit e stattfinden kann
werden durch e beendet

Nachbedingungen : $post(e)$
werden durch e gültig
(evtl. auch:
dürfen nicht gelten, wenn e stattfinden soll)

Evtl. zugelassen: $pre(e) \cap post(e) \neq \emptyset$
(Vorbedingungen, die nicht verändert werden)

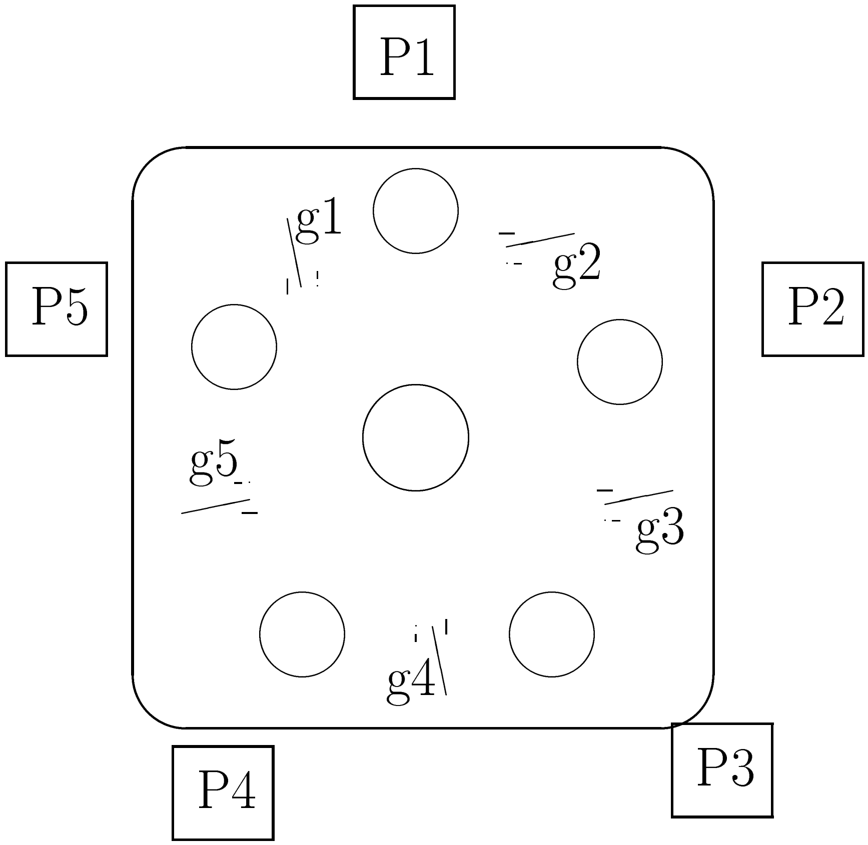
Andere Modellierung:

Vorbedingungen : $pre(e)$
müssen gelten, damit e stattfinden kann

beendete Bedingungen : $delete(e)$
werden durch e ungültig

geschaffene Bedingungen : $add(e)$
werden durch e gültig

Beispiel “5 Philosophen”:



Bedingungen :

`gj` : Gabel g_j liegt auf dem Tisch

`Pigj` : Philosoph P_i hat Gabel g_j

Ereignisse :

`PiR` : Philosoph P_i nimmt rechte Gabel
(Vorbedingung: `gj`, Nachbedingung: `Pjgj`)

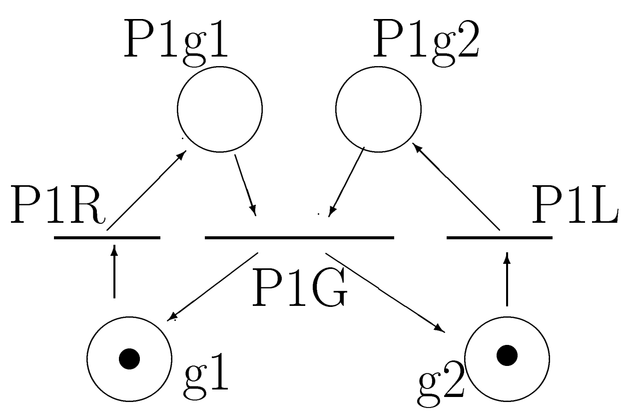
`PiL` : Philosoph P_i nimmt linke Gabel

`PiG` : Philosoph P_i gibt Gabeln zurück

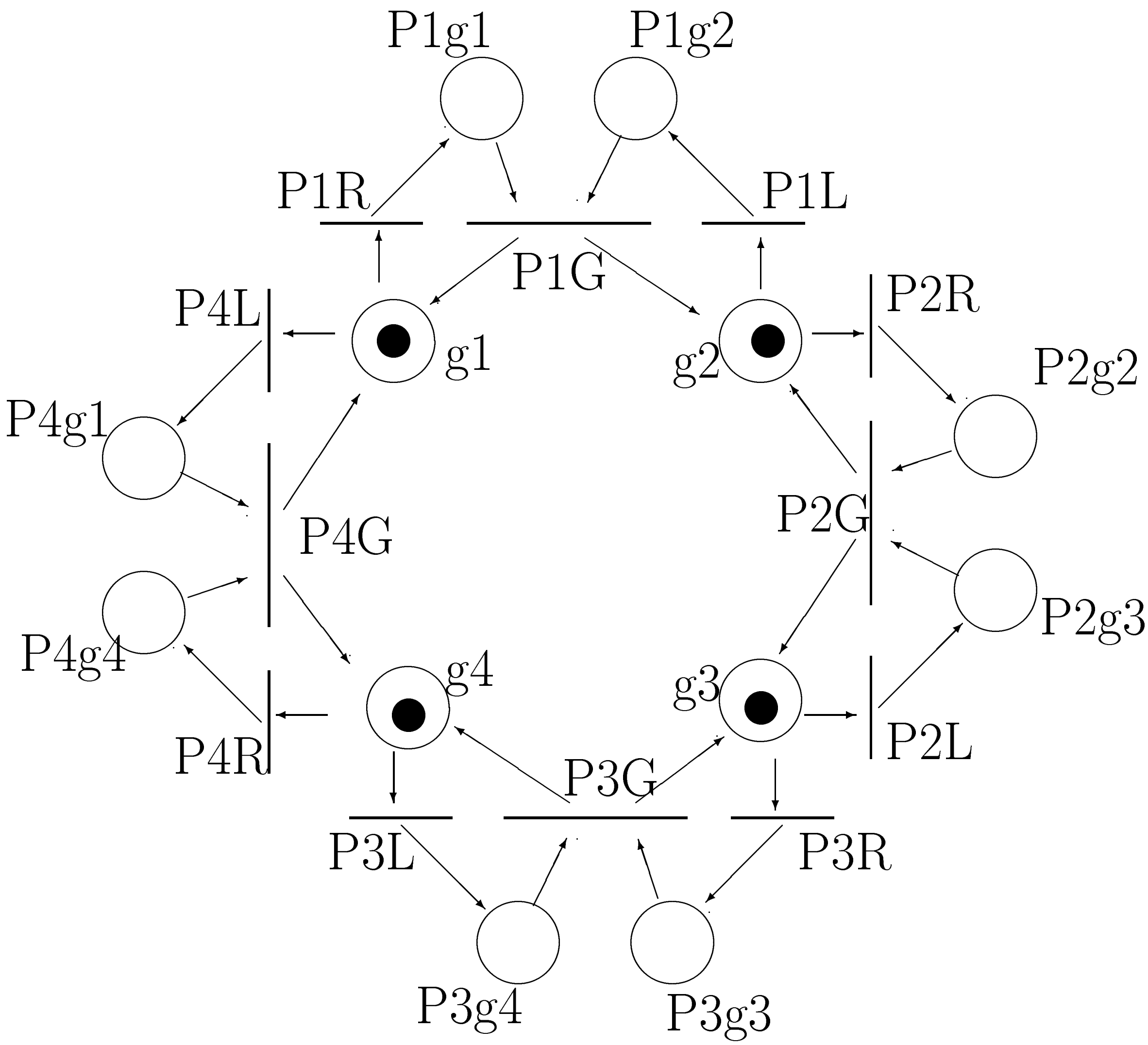
Graphische Darstellung als
"Bedingungs-/Ereignis-Netz"

- Striche für Ereignisse
- Kreise für Bedingungen
- Pfeile von Vorbedingungen
  bzw. zu Nachbedingungen
- Geltende Bedingungen markiert

für 1. Philosoph:



Netz für 4 Philosophen:



Fälle beschrieben als Menge geltender Bedingungen

„*Gabeln auf dem Tisch*“: $\{\texttt{g1}, \texttt{g2}, \texttt{g3}, \texttt{g4}\}$

„*Philosophen 2 und 4 essen*“: $\{\texttt{P2g2}, \texttt{P2g3}, \texttt{P4g4}, \texttt{P4g1}\}$

Spezielle Situationen

Konflikt ,

z.B.

zwischen `P1L` und `P2R` wenn `g2` gilt

Deadlock ,

z.B. Fall

$$\{\texttt{P1g1}, \texttt{P2g2}, \texttt{P3g3}, \texttt{P4g4}\}$$

Livelock (“Starvation”),

z.B.

Wiederholte Abfolge nur von

```
P2R P2L P4R P4L P2G P4G
    P2R P2L P4R P4L P2G P4G
        P2R P2L P4R P4L P2G P4G  .....
```

d.h.: P1 und P3 kommen nicht zum Zuge

9.3 Petri-Netze

Erweiterung: *„Plätze“* statt Bedingungen mit evtl. mehreren *Marken* auf einem Platz

Definition

$N = [P, T, F, V, K, m_0]$ ist ein *Petri-Netz*, falls

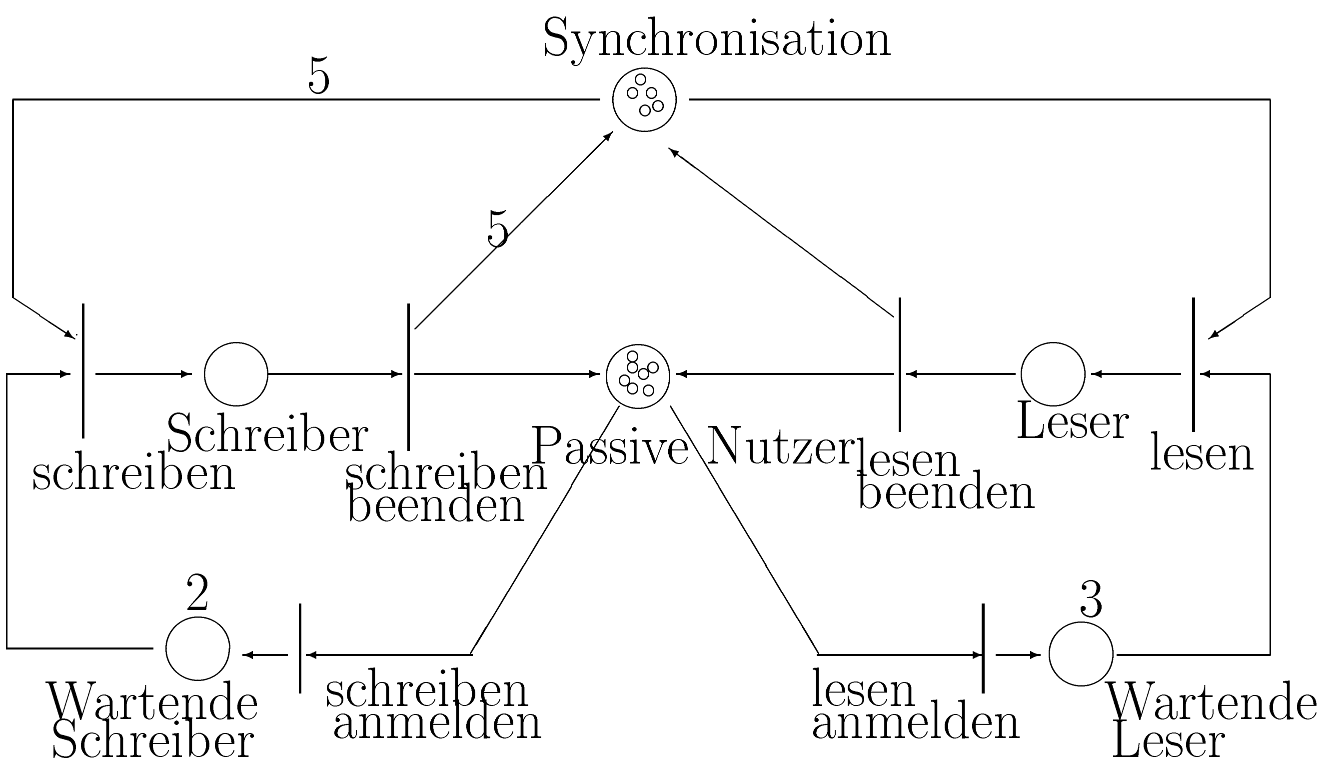
- P ist die Menge der Plätze (oder ”Stellen”), dargestellt durch Kreise
- T ist die Menge der Transitionen, dargestellt durch Striche
- $P \cap T = \emptyset$, $P \cup T \neq \emptyset$
- $F \subseteq P \times T \cup T \times P$ ist die Flußrelation dargestellt durch Pfeile:
  von p nach t für $[p, t] \in F$ bzw.
  von t nach p für $[t, p] \in F$
- $V : F \to \mathcal{N}^+$ ist die Vielfachheit der Bögen aus F, dargestellt durch Beschriftung
- $K : P \to \mathcal{N}^+ \cup \{\infty\}$ ist die Kapazität der Plätze, dargestellt durch Beschriftung
- $m_0 : P \to \mathcal{N}$ (andere Schreibweise: $m \in \mathcal{N}^P$) ist die initiale Markierung der Plätze dargestellt durch Markierung/Beschriftung

Darstellung als bipartiter Graph, Beispiel:

Lesen/geschütztes Schreiben eines Datenspeichers

maximal 5 parallele Lesezugriffe
oder aber 1 Schreibzugriff

maximal 2 bzw. 3 wartende Prozesse



Falls nicht spezifiziert:
Kapazitäten unbeschränkt (∞), Vielfachheiten : 1

Dynamik: Bewegung von Marken

Definition

$N = [P, T, F, V, K, m_0]$ sei ein *Petri-Netz*.
Abkürzende Schreibweise: $V(x, y) := V([x, y])$.

1. Für Transition $t \in T$ werden $t^+, t^- \in \mathcal{N}^P$ und $\Delta t \in \mathcal{Z}^P$ definiert:

   $$t^+(p) := V(t, p)$$

   $$t^-(p) := V(p, t)$$

   (Zahl der Marken, die t auf p legt bzw.von p nimmt)

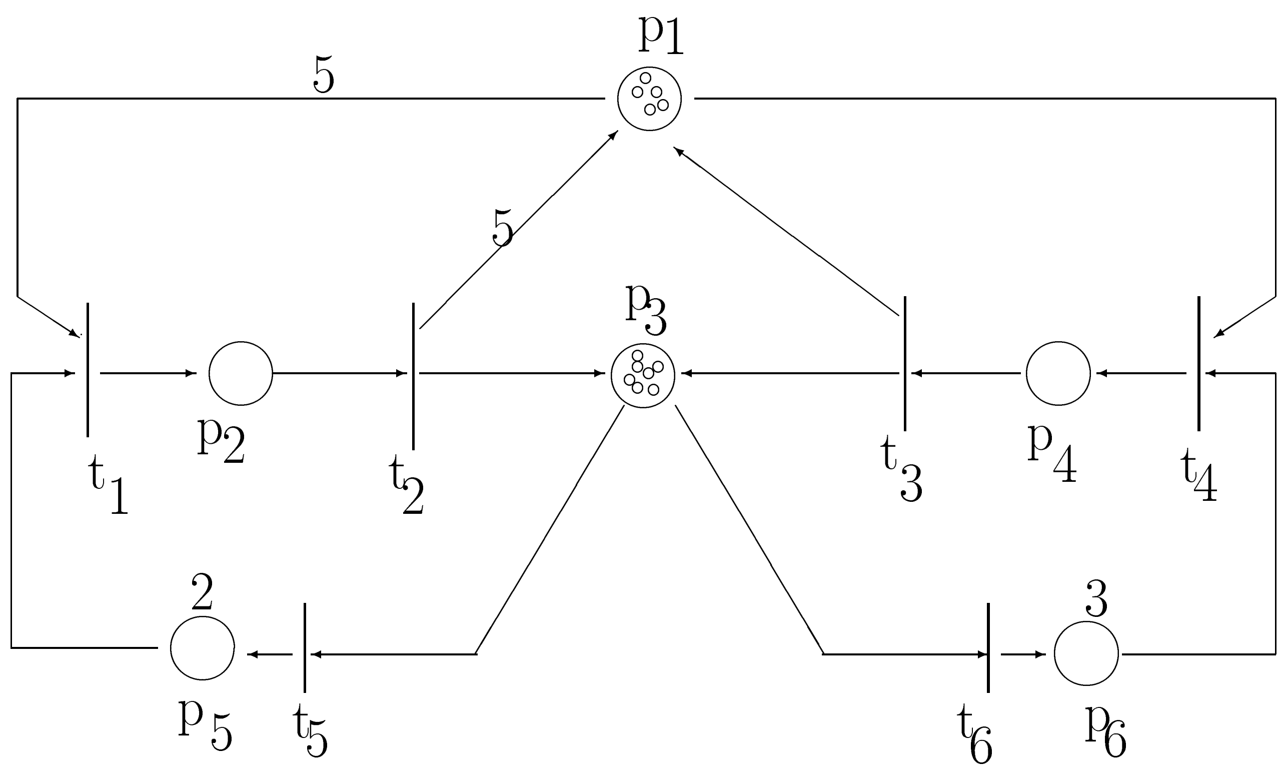
   $$\Delta t(p) := t^+(p) - t^-(p)$$

   (Änderung auf p durch Schalten von t)

2. Transition $t \in T$ hat Konzession (“kann schalten”) bei Markierung $m \in \mathcal{N}^P$, falls

   - $\forall p \in P : t^-(p) \leq m(p)$ (kurz: $t^- \leq m$)
   - $\forall p \in P : m(p) + \Delta t(p) \leq K(p)$

3. Falls Transition $t \in T$ bei Markierung $m \in \mathcal{N}^P$ Konzession hat und schaltet, so ergibt sich die neue Markierung $m + \Delta t$.



$t_1^- = (5, 0, 0, 0, 1, 0)$
$t_1^+ = (0, 1, 0, 0, 0, 0)$

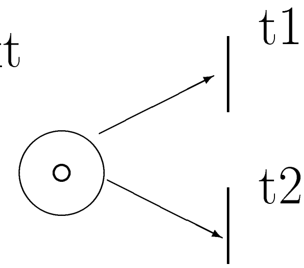
$\Delta t_1 = (-5, 1, 0, 0, -1, 0)$

Bei $m = (5, 0, 3, 0, 1, 3)$ haben Konzession: t_1, t_4, t_5

Schalten von t_4 führt zu $m' = (4, 0, 3, 1, 1, 2)$,
danach haben Konzession: t_3, t_4, t_5, t_6

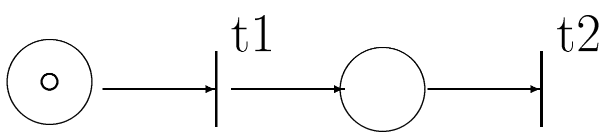
Spezielle Konstrukte

Konflikt

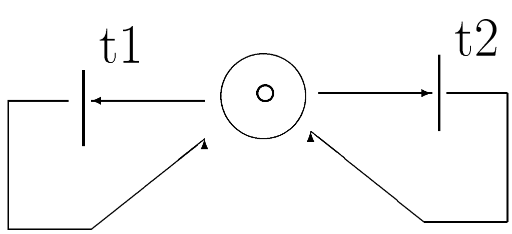


Nebenläufigkeit

Sequentialität



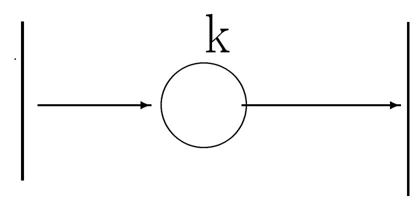
gegenseitiger Ausschluss



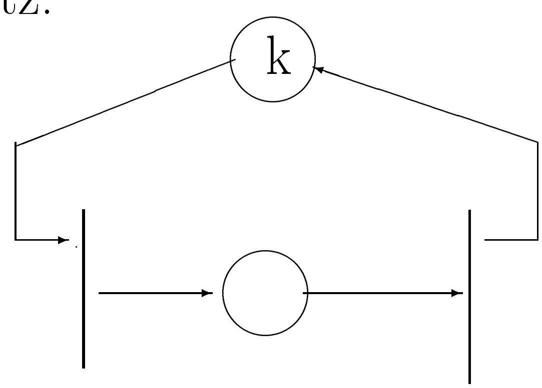
Modellierung von endlichen Kapazitäten durch spezielle Plätze:

Beispiel:

Statt



zusätzlicher Platz:



$\Rightarrow$ Auf Kapazitäten bei Untersuchung verzichten

Schaltfolgen

Definition

$N = [P, T, F, V, K, m_0]$ sei ein *Petri-Netz*.

1. Transitionsfolge $u = t_1 ... t_n \in T^*$ kann schalten bei Markierung $m \in \mathcal{N}^P$,

   falls für alle $i = 1, ..., n$ gilt:
   t_i kann schalten bei $m + \Delta t_1 + \Delta t_2 + ... \Delta t_{i-1}$

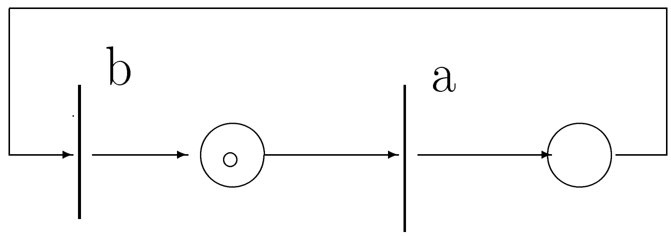
2. Beim Schalten von $u = t_1 ... t_n \in T^*$ in $m \in \mathcal{N}^P$ wird die neue Markierung $m + \Delta u$ erreicht mit

   $$\Delta u := \Delta t_1 + \Delta t_2 + ... \Delta t_n$$

3. Die Menge aller Folgen $u \in T^*$, die bei m_0 schalten können, bilden die "Schaltsprache"

   $$L_N := \{u \mid u \in T^* \wedge u \text{ kann schalten bei } m_0\}.$$

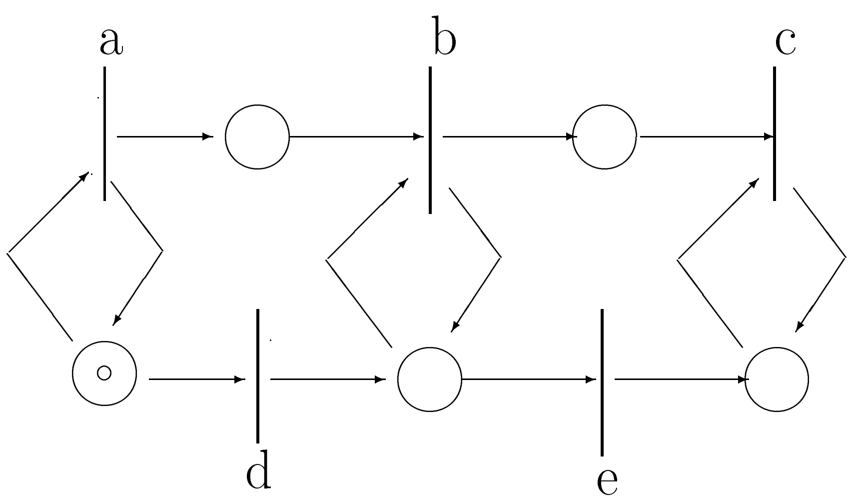
Beispiele:



$L_N = Pref(\{\, ab \,\})$

dabei $Pref(L) := \{u \mid \exists v \in L : u \sqsubseteq v\}$

Bemerkung: $\Delta u \le m$ bedeutet nicht,
daß u bei m schalten kann,
z.B. $ba \in L_N$, obwohl $\Delta ba = (0,0)$.



$L_N = Pref(\{\, a^k d b^m e c^n \mid k \ge m \ge n \,\})$
(nicht kontextfrei!)

Erreichbarkeit

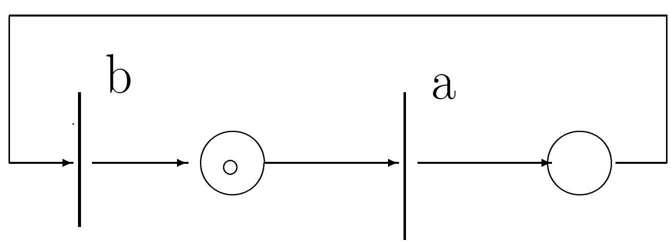
Definition

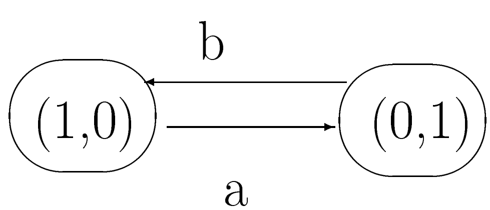
$N = [P, T, F, V, K, m_0]$ sei ein *Petri-Netz*.

1. m' ist erreichbar von m, falls $u \in T^*$ existiert mit u kann in m schalten und $m' = m + \Delta u$.

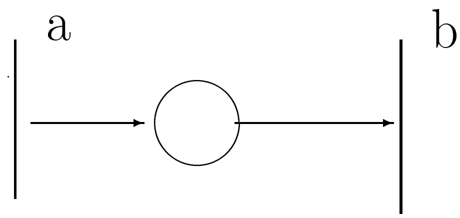
2. Erreichbarkeitsmenge (von m_0 aus):
$$R_N := \{m_0 + \Delta u \mid u \in L_N\}.$$

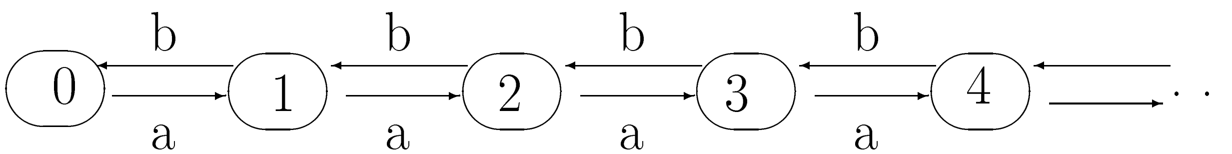
3. Erreichbarkeitsgraph:
   - Knoten: R_N
   - Kanten:
     mit t markierte Kante von $m \in R_N$ nach $m+\Delta t$, falls t Konzession hat in m





Auch für endliche N kann R_N unendlich sein

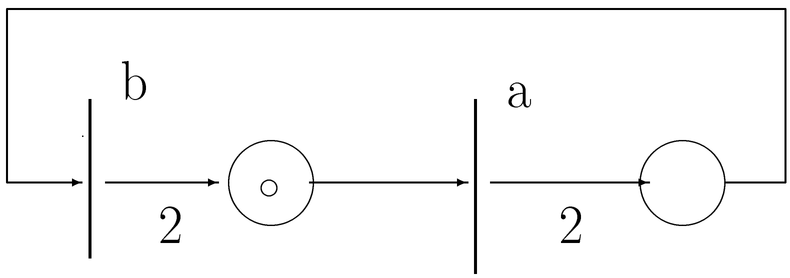


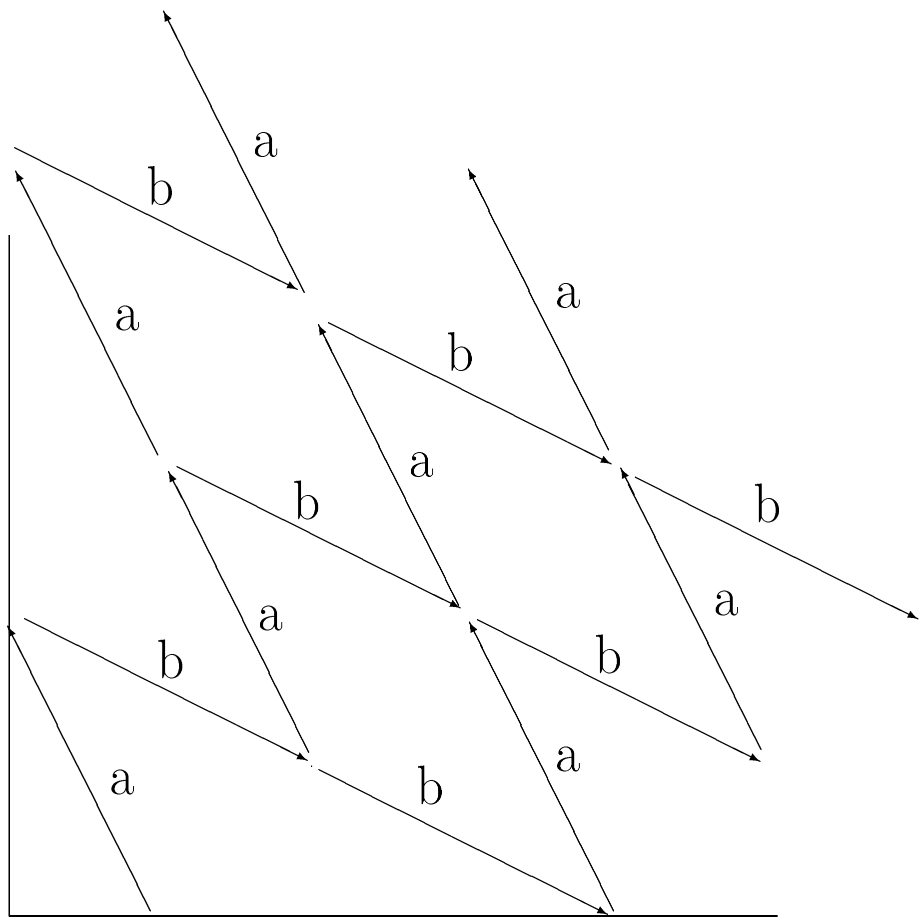


Struktur von R_N

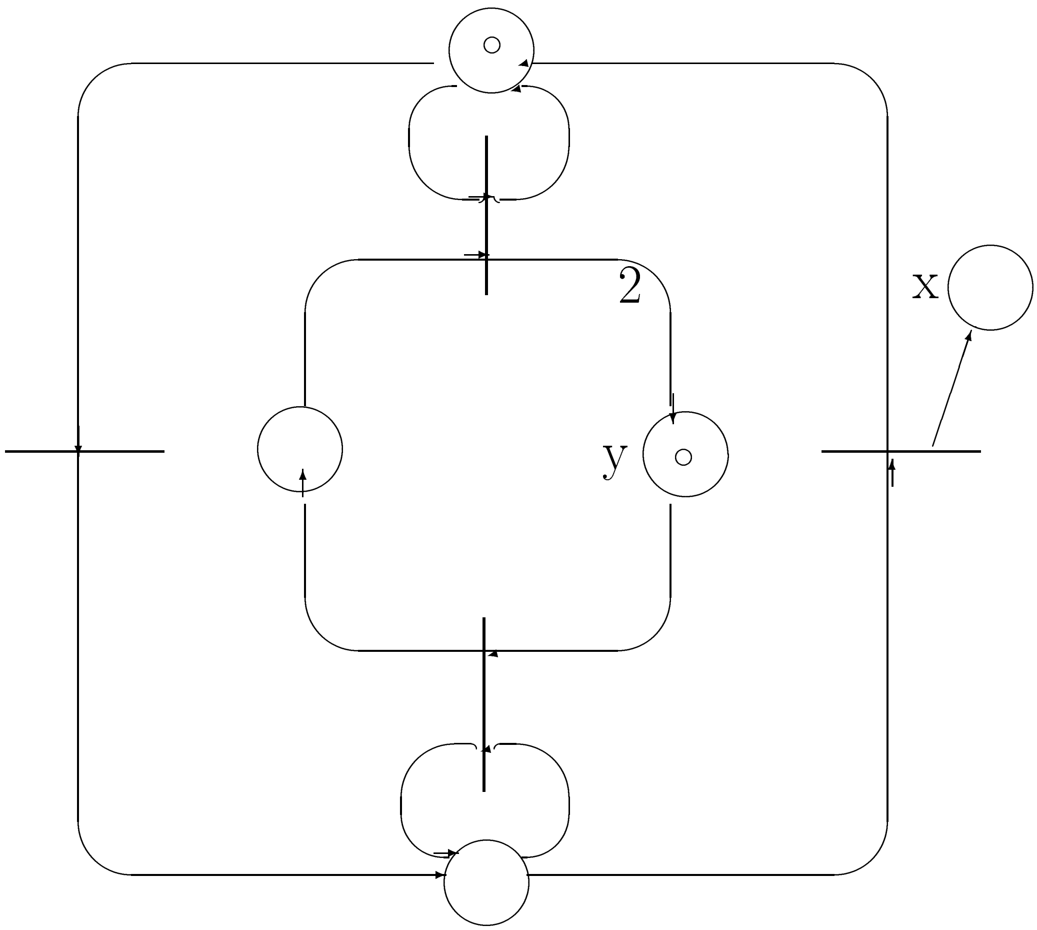
R_N als Menge von Punkten in $\mathcal{N}^P$,

Transitionen t bewirken lineare Übergänge um Δt.





Beispiel für nicht-lineare Menge R_N



Projektion von R_N für Plätze x und y:

$$\{(x, y) \mid x \in \mathcal{N} \wedge y \leq 2^x\}$$

Analyse von Petri-Netzen

- Lebendigkeit:
  kann eine Transition stets wieder zum Schalten gebracht werden
- Sicherheit:
  Schranken für Markierung auf den Plätzen
- Erreichbarkeit:
  ist eine bestimmte Markierung erreichbar
- Reproduzierbarkeit:
  ist eine bestimmte Markierung stets wieder erreichbar
- Deadlock-Freiheit
- Livelock-Freiheit
- ...

Methoden:

- Strukturelle Eigenschaften
- Simulation
- Lineare (Un-)Gleichungssysteme ("Invarianten")
- ...