Sommersemester 2017

Petrinetze Teil 2

Übung
„Modellierung und Spezifikation“

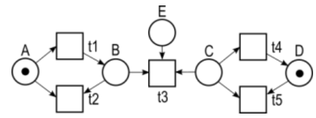
Robert Prüfer
pruefer@informatik.hu-berlin.de

Theory of Programming

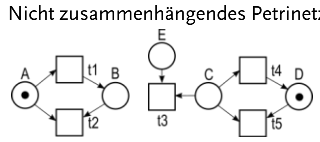
Zusammenhängende Netze

Ein Netz ist zusammenhängend gdw. sein zugrundeliegender ungerichteter Graph zusammenhängend ist.

- Zusammenhängendes Petrinetz:



- Nicht zusammenhängendes Petrinetz:



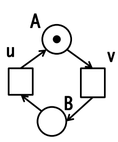
2

Beschränktheit von Netzen

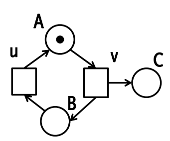
- Obere Grenze k für Marken
  auf jedem Platz
  in jeder erreichbaren Markierung

- Gegenteil: unbeschränkt

beschränktes Netz:



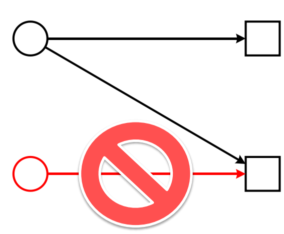
unbeschränktes Netz:



3

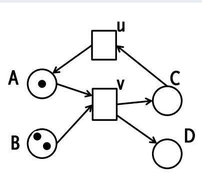
Free-Choice-Eigenschaft eines Petrinetzes

- Jeder geteilte Vorplatz ist der einzige Vorplatz seiner Nachtransitionen



4

Platzinvarianten



- Invarianz: „etwas bleibt gleich“
- Platzinvariante: Summe von Marken auf Plätzen ist invariant
- darstellbar durch Vektoren

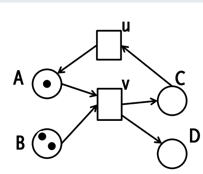
A	B	C	D
1	0	1	0
0	1	0	1
1	1	1	1
9	0	9	0

Jede Zeile beschreibt eine Platzinvariante.

- Invarianten in TINA: „semiflows“

5

Gleichungen aus Platzinvarianten

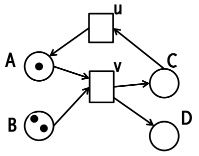


- Anfangsmarkierung auswerten
  - Anfangsmarkierung erzeugt rechte Seite der Gleichung

A	B	C	D		
1	0	1	0	→	A + C = 1
0	1	0	1	→	B + D = 2
1	1	1	1	→	A + B + C + D = 3
9	0	9	0	→	9A + 9C = 9

6

Minimale Invarianten

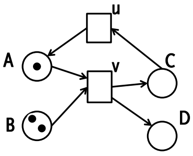


A	B	C	D
1	0	1	0
0	1	0	1
~~1~~	~~1~~	~~1~~	~~1~~
~~9~~	~~0~~	~~9~~	~~0~~

- Minimalität
  - Eine Invariante ist minimal, wenn sie sich nicht aus anderen kombinieren lässt

7

Ist ein Vektor eine Platzinvariante?



Inzidenzmatrix $\underline{N}$

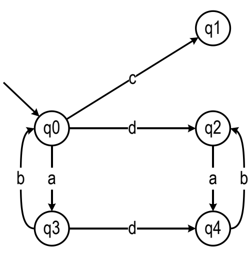
Platzinvariante

$$\begin{pmatrix}1 & 0 & 1 & 0\end{pmatrix} \bullet \begin{array}{c c} & \begin{array}{cc} \mathtt{u} & \mathtt{v} \end{array} \\ \begin{array}{c} \mathtt{A} \\ \mathtt{B} \\ \mathtt{C} \\ \mathtt{D} \end{array} & \begin{pmatrix} 1 & -1 \\ 0 & -1 \\ -1 & 1 \\ 0 & 1 \end{pmatrix} \end{array} = \begin{pmatrix}0 & 0\end{pmatrix}$$

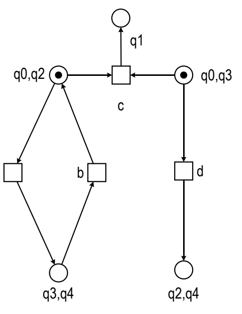
8

Das Syntheseproblem

geg.: Zustandsmaschine

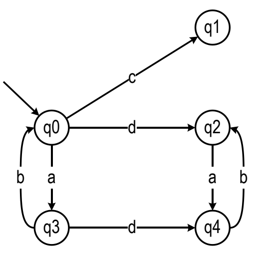


ges.: Petrinetz



9

Regionen



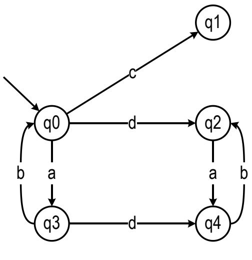
- *R* ist eine Region, wenn für **jede** Kantenbeschriftung *x* gilt:
  - *R* **empfängt jede oder keine** *x*-Kante

  UND

  - *R* **verschickt jede oder keine** *x*-Kante.

10

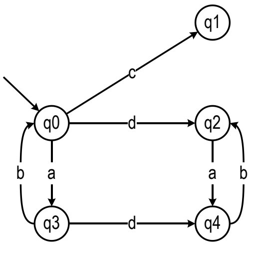
Regionen



- Beispiele für Regionen:
  - q1
  - q1,q2,q4
  - q0,q3
  - …

11

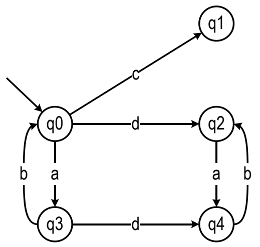
Triviale Regionen



- {}
- q0,q1,q2,q3,q4

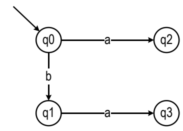
12

Minimale Regionen



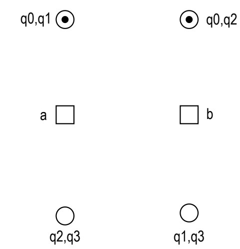
- Minimalität bezüglich Teilmenge
  - q1
  - q2,q4
  - ~~q1,q2,q4~~

Synthese – Teil 1

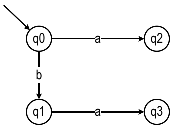


Regionen:
{q0,q1}, {q0,q2}, {q1,q3}, {q2,q3}

1. Finde jede nicht-triviale, minimale Region *R*.
2. Füge für jedes *R* einen Platz *R* ein.
3. Füge für jede Kantenbeschriftungen *x* eine Transition *x* ein.



Synthese – Teil 2



1. **Verschickt** *R* ein *x*, füge Kante von *R* nach *x* ein.
2. **Empfängt** *R* ein *x*, füge Kante von *x* nach *R* ein.
3. **Enthält** *R* alle *x*, füge Schlinge zwischen *R* und *x* ein.