

# Software Engineering in der Praxis

## Praktische Übungen

# Zeitbehaftete Petrinetze

Florin Pinte   Marc Spisländer

Lehrstuhl für Software Engineering  
Friedrich-Alexander-Universität Erlangen-Nürnberg

- 1 Inhalt
  
- 2 Nachlese
  - Lernziele der letzten Übung
  
- 3 Zeitbehaftete Petrinetze
  - Überblick
  - Analysierbarkeit
  
- 4 Aufgaben



# Analyse und Simulation mit Petrinetzen

- Formales Beschreibungsmittel für Nebenläufigkeit und Synchronisation
- Fachvokabular
  - Lebendigkeit, Verklemmungen
  - Erreichbarkeit
  - Kontakte, Konflikte
- Petrinetz-Schablonen
- Analyse und Simulation mit Software

# Zeitbehaftete Petrinetze

Erweiterung von einfachen Petrinetzen um die Zeitaspekte

- Frühester Schaltzeitpunkt nach Aktivierung (»Earliest Firing Time«, EFT)
- Spätester Schaltzeitpunkt nach Aktivierung (»Latest Firing Time«, LFT)

einer Transition.

# Schaltintervalle

Spezifikation im Petrinetz als *Statisches Schaltintervall*

- $S : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$   
also z. B.  $S(t_1) = [2, \infty[$

Zuordnung zur Laufzeit als *Dynamisches Schaltintervall*

- $D : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$

Projektoren:  $S(t) = [s_1(t), s_2(t)]$  und  $D(t) = [d_1(t), d_2(t)]$

Zeitbehaftete Petrinetze sind eine Verallgemeinerung der klassischen Petrinetze.  $S$  für klassische Petrinetze =  $[0, \infty)$

# Schaltintervalle

Spezifikation im Petrinetz als *Statisches Schaltintervall*

- $S : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$   
also z. B.  $S(t_1) = [2, \infty[$

Zuordnung zur Laufzeit als *Dynamisches Schaltintervall*

- $D : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$

Projektoren:  $S(t) = [s_1(t), s_2(t)]$  und  $D(t) = [d_1(t), d_2(t)]$

Zeitbehaftete Petrinetze sind eine Verallgemeinerung der klassischen Petrinetze.  $S$  für klassische Petrinetze =  $[0, \infty)$

# Schaltintervalle

Spezifikation im Petrinetz als *Statisches Schaltintervall*

- $S : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$   
also z. B.  $S(t_1) = [2, \infty[$

Zuordnung zur Laufzeit als *Dynamisches Schaltintervall*

- $D : T \rightarrow \mathbf{Q}_0 \times (\mathbf{Q}_0 \cup \infty)$

Projektoren:  $S(t) = [s_1(t), s_2(t)]$  und  $D(t) = [d_1(t), d_2(t)]$

Zeitbehaftete Petrinetze sind eine Verallgemeinerung der klassischen Petrinetze.  $S$  für klassische Petrinetze =  $[0, \infty)$



# TPN-Schaltbereitschaft

Wird  $t$  wird zum Zeitpunkt  $\tau_{abs}$  PN-schaltbereit, d. h.

- unmittelbar zuvor waren nicht alle Eingangsplätze besetzt,
- sie sind es aber zum Zeitpunkt  $\tau_{abs}$

dann ist die Transition  $t$  im Zeitraum  $[\tau_{abs} + s_1(t), \tau_{abs} + s_2(t)]$  TPN-schaltbereit, wenn die *PN* Schaltbereitschaft

- zu keinem Zeitpunkt unterbrochen und
- durch keinen Schaltvorgang einer im Konflikt stehenden Transition gestört

wurde.

Formal: siehe Grundlagenvorlesung.

# Zuweisung des dynamischen Schaltintervalls

- Transition ist nicht PN-schaltbereit:  $D = \emptyset$
- Transition wird zum Zeitpunkt  $\tau_{abs}$  PN-schaltbereit:  
 $d_1(t) = s_1(t); d_2(t) = s_2(t)$
- Transition ist nach  $\tau_{abs} + \tau$  immer noch PN-schaltbereit:  
 $d_1(t) = \max\{0, s_1(t) - \tau\}; d_2(t) = s_2(t) - \tau$

$\tau$  ist dabei die Stoppuhr, die zum Zeitpunkt  $\tau_{abs}$  zu laufen beginnt und mit dem Schaltzeitpunkt, spätestens jedoch mit  $\tau_{abs} + s_2(t)$  stoppt.

# Zustände von TPNs

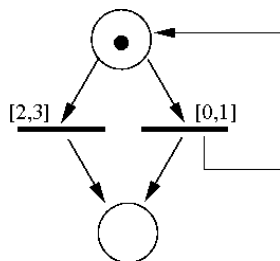
Der aktuelle Zustand eines zeitbehafteten Petrinetzes (TPN) ist nun ein Paar  $(M, D)$  bestehend aus

- der Markierung  $M$  des Petrinetzes
- und dem Vektor  $D$  der aktuellen Schaltintervalle

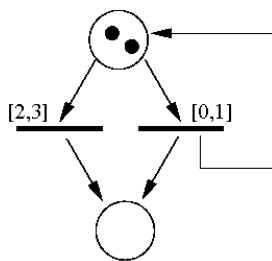
# Zeitbehaftete Petrinetze

## Analysierbarkeit

- *TPNs* sind komplexer und ausdrucksstärker als einfache Petrinetze, dafür aber **nicht so gut analysierbar**.
- Analyse der Erreichbarkeit und Beschränktheit i. A. unentscheidbar, oft jedoch möglich.



**Konfliktsituation**



**kein Konflikt**

# Zustandsklassen

Menge der Nachfolger-Zustände im Allgemeinen unendlich. Daher Bildung von *Zustandsklassen*  $C = (M, D^*)$  mit

- einer Markierung  $M$
- der Vereinigung  $D^*$  der dann möglichen dynamischen Schaltintervalle jeder Transition

durch eine rekursive Prozedur.

# Werkzeug und Aufgabe

- TINA/LAAS  
*Laboratoire d'Analyse et d'Architecture des Systèmes*
- Bedienung: »effizient« bis eigenwillig

# TINA Benutzung

## Editiermodus

Grafischer Editiermodus für *Time Petri Nets*:

- Strg + Linke Maustaste: Auf Zeichenfläche bewegen
- Hochstell + Linke Maustaste: Zeichenfläche zoomen
- Strg + g: Zeichenraster einblenden
- Mittlere Maustaste: Stelle erzeugen
- Strg + Mittlere Maustaste: Transition erzeugen
- Mittlere Maustaste + ziehen: Relation erzeugen
- Rechte Maustaste auf Element: Eigenschaften ändern

Alle Bedientechniken sind im Menüpunkt »help« beschrieben.



# TINA Benutzung

## Analysefunktionen

- **Reachability:** Hier empfiehlt sich die Analyse des Zustandsklassengraphen (*marking graph*). Als Ausgabeformat eignet sich zur manuellen Interpretation das Format “verbose”.
- **Stepper Simulation:** Zeitfortschritt kann manuell gesteuert werden. Bei den Transitionen wird das dynamische Zeitintervall angezeigt und können, wenn rot markiert, geschaltet werden.