

Petri-Netze, Anwendungen und Tools

Autor: Roland Meyer

Betreuer: Hans Fleischhack

`Roland.Meyer@informatik.uni-oldenburg.de`

Überblick

- Low-Level Petri-Netze
- Eigenschaften von Petri-Netzen
- High-Level Petri-Netze
- Petri-Box Kalkül
- Modellierung mit Petri-Netzen
- Verifikation mit Petri-Netzen
- Fazit

Grundlagen der Low-Level Petri-Netze

- Was ist ein Low-Level Petri-Netz?
- Was ist ein Zustand?
- Wie schaltet eine Transition?
- Was ist der Erreichbarkeitsgraph?

Beispiel: Modellierung eines Geschäftsprozesses

Sicherheitseigenschaften

- Eine Stelle ist *sicher*, falls sie in jedem Zustand höchstens ein Token trägt.
- Eine Stelle ist *k-beschränkt* ($k \in \mathbb{N}$), falls sie in jedem Zustand höchstens k Token trägt.
- Eine Stelle ist *beschränkt*, falls es ein $k \in \mathbb{N}$ gibt, so dass sie k -beschränkt ist.
- Ein Netz ist *sicher*, *k-beschränkt*, *beschränkt*, falls dies für alle Stellen gilt.

Lebendigkeitseigenschaften

- Ein Netz ist *deadlockfrei*, falls jede erreichbare Markierung eine Folgemarkierung hat.
- Eine Transition ist *lebendig*, falls sie immer wieder geschaltet werden kann.
- Eine Netz ist *lebendig*, falls dies für alle Transitionen gilt.

Entscheidung der Eigenschaften

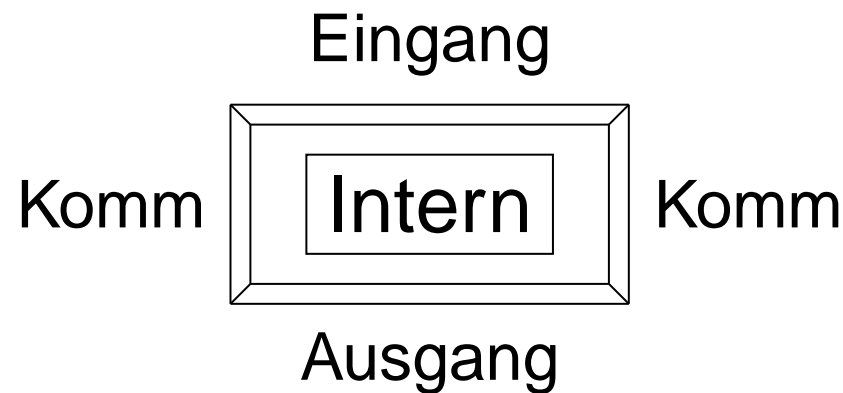
- Entscheidung am endlichen Erreichbarkeitsgraphen
- Unendliche Erreichbarkeitsgraphen
- Idee des Überdeckungsgraphen
- Konstruktion des Überdeckungsgraphen
- Entscheidung am Überdeckungsgraphen

High-Level Netze

- Stellen besitzen einen Datentyp
- Token sind Daten-Werte
- Kanten sind mit Termen beschriftet, Variablen sind getypt
- Transitionen sind mit Booleschen Ausdrücken beschriftet
- Transitionen sind mit Aktionssymbolen beschriftet

Der Petri-Box Kalkül

- Idee des Petri-Box Kalküls
- Was ist eine Petri-Box?



- Operatoren zur vertikalen Komposition
- Operatoren zur horizontalen Komposition

Vertikale Komposition

- Sequentielle Komposition $N_1; N_2$: Erst N_1 , dann N_2
- Auswahlkomposition $N_1 [] N_2$: Entweder N_1 oder N_2
- Parallelkomposition $N_1 \parallel N_2$: N_1 parallel zu N_2
- Iteration $[N_1 * N_2 * N_3]$: Initialisierung mit N_1 , Ausführung von N_2 , Terminierung durch N_3

Horizontale Komposition

- Synchronisation $N \text{ sy } A$: Synchronisation auf Aktion A
- Restriktion $N \text{ rs } A$: Ausblenden von A -Transitionen
- Scoping $[A : N]$: Synchronisiere auf A und blende dann A -Transitionen aus

Petri-Netz Semantik für $B(PN)^2$

- Daten-Werte
- Wertzuweisung
- Bedingte Anweisungen
- Schleifen-Konstrukte
- Parallel-Komposition

Verifikation mittels Petri-Netzen

- Erreichbarkeitsgraphen und Kripke-Strukturen
- LTL Model-Checking
- Prozesse, Entfaltung und das vollständige, endliche Präfix
- AL+Possibly Model-Checking
- Co-lineare Eigenschaften
- Semi-Entscheidung co-linearer Eigenschaften

Fazit und Ausblick

- Leicht verständlich
- Strikt mathematisches Modell - Lineare Algebra
- Vielfältige Einsatzmöglichkeiten
- Erlaubt Model-Checking
- Zeitbehaftete Petri-Netze
- Spezielle Netzklassen
- Strukturelle Eigenschaften

References

- [Eik93] Eike Best, Richard P. Hopkins. $B(PN)^2$ - a Basic Petri Net Programming Notation. *PARLE*, 1993.
- [Han96a] Hans Fleischhack. Skizzen und Notizen zur Vorlesung Netzsemantik für verteilte Systeme, 1996.
- [Han96b] Hans Fleischhack, Bernd Grahlmann. A Petri Net Semantics for $B(PN)^2$ with Procedures which Allows Verification. Technical report, University of Oldenburg, 1996.
- [ISO00] High-Level Petri Nets - Concepts, Definitions and Graphical Notation, October 2000. Final Draft International Standard.
- [Jav96] Javier Esparza, Stefan Römer, Walter Vogler. An Improvement of McMillan's Unfolding Algorithm. *Tools and Algorithms for Construction and Analysis of Systems*, 1996.
- [Lut03] Lutz Priese, Harro Wimmel. *Petri-Netze*. Springer-Verlag, 2003.