

# FGI-2 – Formale Grundlagen der Informatik II

## Modellierung und Analyse von Informatiksystemen

### Präsenzlösung 6: Asynchrone Produkte, part. Ordnung, Nachrichtenordnung, Arbeiten mit RENEW

Präsenzteil am 16./17. 11. – Abgabe am 23./24. 11. 2015

**Präsenzaufgabe 6.1:** Seien  $\leq$ ,  $R$ ,  $R_1$  und  $R_2$  innere Relationen über derselben Basismenge  $A$ .

1. Sei  $\leq$  eine partielle Ordnung. Ist  $R = (\leq \cup \leq^{-1})$  eine Äquivalenzrelation?

**Lösung:** Gegenbeispiel: Sei  $a \leq b$  und  $a \leq c$ . Dann ist  $R = \{(a, b), (b, a), (a, c), (c, a)\} \cup id$  nicht transitiv, da  $(b, a), (a, c) \in R$  gilt, aber nicht  $(b, c)$ .

2. Zeige: Seien  $R_1$  und  $R_2$  partielle Ordnungen, dann ist  $R_1 \cap R_2$  ebenfalls eine partielle Ordnung.

**Lösung:**  $R = R_1 \cap R_2$  ist ebenfalls eine partielle Ordnung.

Reflexivität ist klar: Da  $aR_i a, i = 1, 2$  für alle  $a \in A$  gilt, ist auch  $aRa$ .

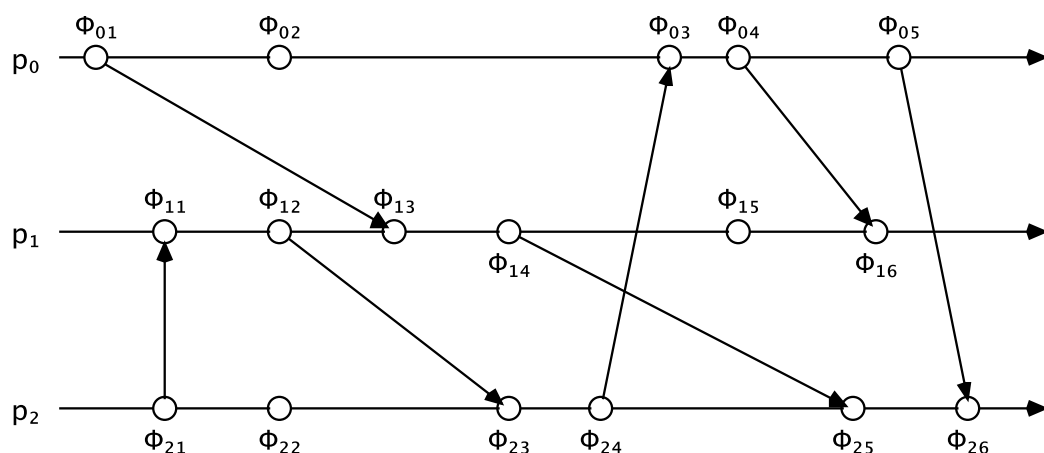
Antisymmetrie: Sei  $aRb, bRa, i = 1, 2$ , dann muss  $aR_i b, bR_i a, i = 1, 2$  gewesen sein und da  $R_i$  antisymmetrisch ist, gilt auch  $a = b$ .

Transitivität. Sei  $aRb, bRc, i = 1, 2$ , dann muss  $aR_i b, bR_i c, i = 1, 2$  gewesen sein und deswegen gilt auch  $aR_i c, i = 1, 2$ , also auch  $aRc$ .

3. Zeige: Seien  $R_1$  und  $R_2$  partielle Ordnungen, dann ist  $R_1 \cup R_2$  i.a. keine partielle Ordnung.

**Lösung:** Gegenbeispiel: Sei  $R_1 = \{(a, b)\} \cup id$  und  $R_2 = \{(b, c)\} \cup id$ . Dann ist  $R := R_1 \cup R_2 = \{(a, b), (b, c)\} \cup id$  nicht transitiv, da  $(a, b), (b, c) \in R$  gilt, aber nicht  $(a, c)$ .

4. Gegeben sei das folgende Zeitskalenmodell. Geben Sie  $LT(\phi_i)$  für alle Ereignisse  $\phi_i$  an.



**Lösung:**

$\Phi_{01} : 1, \Phi_{02} : 2, \Phi_{03} : 6, \Phi_{04} : 7, \Phi_{05} : 8,$   
 $\Phi_{11} : 2, \Phi_{12} : 3, \Phi_{13} : 4, \Phi_{14} : 5, \Phi_{15} : 6, \Phi_{16} : 8,$   
 $\Phi_{21} : 1, \Phi_{22} : 2, \Phi_{23} : 4, \Phi_{24} : 5, \Phi_{25} : 6, \Phi_{26} : 9$

5. Ist die Relation  $\cdot\text{vor}\cdot$  i.a. eine strikte Ordnung? Eine totale strikte Ordnung?

**Lösung:** Die Relation  $\cdot\text{vor}\cdot$  (Def. 5.7) ist eine strikte, aber keine totale strikte Ordnung. Für Striktordnungen muss Irreflexivität und Transitivität gelten (Def. 5.2 b), der entsprechende Beweis für die Relation  $\cdot\text{vor}\cdot$  ist auf S. 79 im Skript nachzulesen. Vollständigkeit (Def. 5.2 c.2) gilt nicht, da z.B. die ersten Ereignisse zweier Prozesse, welche nicht sofort eine Nachricht austauschen, voneinander unabhängig sind und somit nicht in der Relation  $\cdot\text{vor}\cdot$  stehen.

6. Warum gilt  $\phi_1 \text{vor} \phi_2 \implies LT(\phi_1) < LT(\phi_2)$ ?

**Lösung:** Stehen zwei Ereignisse aufgrund von Bedingung Def. 5.7 a) in Relation (sie sind Teil desselben Prozesses), so erhält das zweite Ereignis einen höheren Zeitstempel, da jedes Ereignis einen mindestens „um 1 größeren Wert“ (S. 81) als das vorhergehende Ereignis desselben Prozesses erhält.

Stehen zwei Ereignisse aufgrund von Bedingung Def. 5.7 b) in Relation (Sende- und Empfangereignis einer Nachricht), so erhält das zweite mindestens einen um 1 größeren Zeitstempel als der in der Nachricht übertragene, welcher vom Sendeereignis stammt.

Stehen zwei Ereignisse transitiv in Relation (Def. 5.7 c), so gilt, dass alle Zwischenereignisse aufgrund von Bedingung a) oder b) bereits höhere Zeitstempel erhalten haben. Also hat auch das letzte Ereignis der transitiven Kette einen höheren Zeitstempel, somit gilt die Behauptung (siehe Satz 5.12).

7. Warum gilt aber die Umkehrung  $LT(\phi_1) < LT(\phi_2) \implies \phi_1 \text{vor} \phi_2$  nicht?

**Lösung:** Gegenbeispiel in der Lösung zu 1. Aufgabe:  $\phi_{22}$  und  $\phi_{12}$  stehen nicht in Relation trotz  $LT(\phi_{22}) = 2 < 3 = LT(\phi_{12})$ .

8. Was ändert sich an Teilfragen 6 und 7, wenn wir mit vektoriellen Zeitstempeln arbeiten?

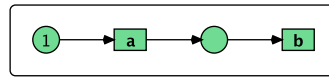
**Lösung:**  $\phi_1 \text{vor} \phi_2 \implies VC(\phi_1) < VC(\phi_2)$  gilt weiterhin, die Argumentation läuft analog zu  $LT$ . Bei jedem Ereignis wird das komponentenweise Maximum der eingehenden Zeitstempel gebildet und um 1 in der zu  $p_i$  gehörenden Komponente vergrößert.

$VC(\phi_1) < VC(\phi_2) \implies \phi_1 \text{vor} \phi_2$  gilt nun auch, da die vektoriellen Zeitstempel von Ereignissen, welche nicht in Relation stehen, ebenfalls unvergleichbar sind (Satz 5.18).

**Präsenzaufgabe 6.2:** Verwenden Sie für diese Übungsaufgabe RENEW auf ihrem Rechner (wenn Sie einen dabei haben).

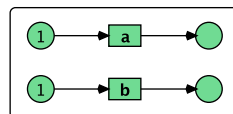
1. Zeichnen Sie ein Petrinetz in dem zwei sequentielle Transitionen a und b vorkommen.

**Lösung:** Ein mögliches Netz:



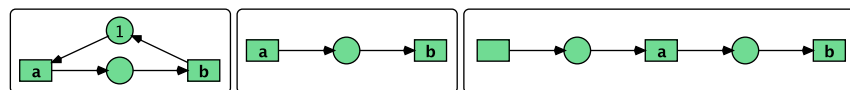
2. Zeichnen Sie ein Petrinetz in dem zwei nebenläufige Transitionen a und b vorkommen.

**Lösung:** Ein mögliches Netz:



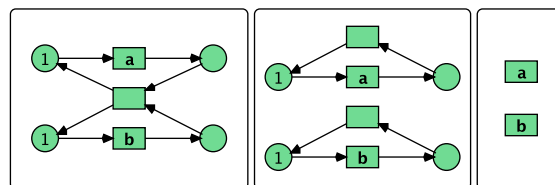
3. Zeichnen Sie ein Petrinetz in dem zwei sequentielle Transitionen a und b immer wieder vorkommen können.

**Lösung:** Drei mögliche Netze:



4. Zeichnen Sie ein Petrinetz in dem zwei nebenläufige Transitionen a und b immer wieder vorkommen können.

**Lösung:** Drei mögliche Netze:

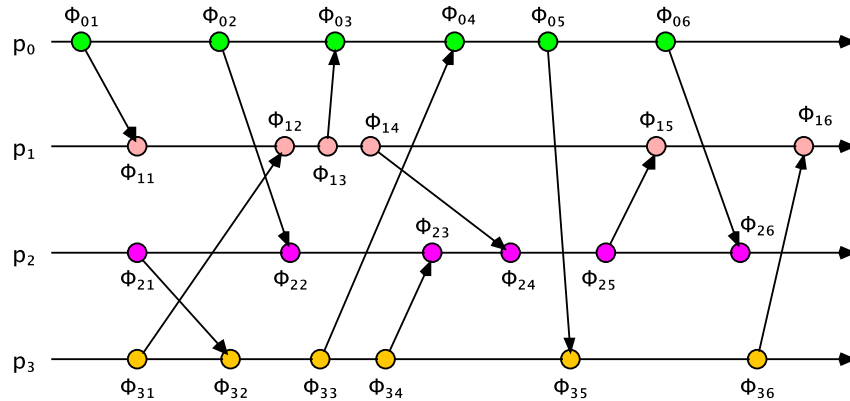


5. Wählen Sie eine geeignete Markierung und simulieren Sie diese Netze mittels RENEW.

**Lösung:** Die Markierung ist in den oberen Netzen bereits mit angegeben.

**Übungsaufgabe 6.3:** Betrachten Sie das folgende Zeitskalenmodell.

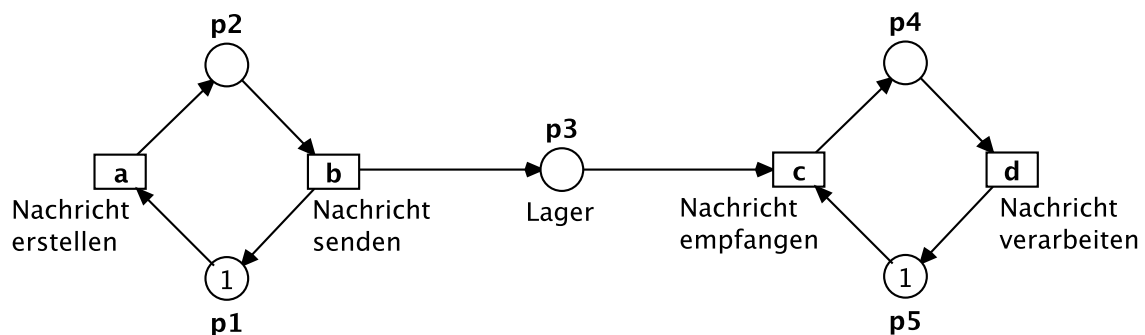
von
6



1. Geben Sie  $VC(\phi_i)$  für alle  $\phi_i$  an.
2. Geben sie für jeden der vier Prozesse jeweils ein Ereignis an, so dass die vier Ereignisse paarweise durch die Relation **vor** angeordnet sind (Def. 5.7). Geben Sie dafür die vektoriellen Zeitstempel an.
3. Geben Sie für jeden der vier Prozesse jeweils ein Ereignis an, so dass die vier Ereignisse paarweise unabhängig sind (Def. 5.16). Geben Sie dafür auch die entsprechenden Vektorzeiten an.
4. Geben Sie für das Zeitskalenmodell für alle Prozessoren die Ereignisse der ersten logischen Zeitstempel bis inklusive  $(3, p_i)$  an. Sie können dabei für die Anordnung statt der Tupel  $(t, p_i)$  die eindeutige Ereignisbezeichnung verwenden.
5. Erstellen Sie für das Zeitskalenmodell einen Präzedenzgraph. Interpretieren Sie sowohl die Abhängigkeiten auf der Zeitachse der Prozessoren als auch die Abhängigkeiten zwischen den Ereignissen der verschiedenen Prozessoren als Präzedenzen.
6. Erstellen Sie für das Zeitskalenmodell ein Petrinetz mittels RENEW. Interpretieren Sie sowohl die Abhängigkeiten auf der Zeitachse der Prozessoren als auch die Abhängigkeiten zwischen den Ereignissen der verschiedenen Prozessoren als kausal geordnet.

**Übungsaufgabe 6.4:** Das folgende P/T-Netz  $N_{6.1} = (P, T, F, W, \mathbf{m}_0)$  stellt den Versand von Nachrichten von einem Sender zu einem Empfänger über ein Lager dar:

von
4



1. Zeichnen Sie dieses Netz mit RENEW (im P/T-Netz-Formalismus!).
2. Ändern Sie das Netz so, dass die Lagerstelle  $p3$  verfeinert wird. Die Verfeinerung soll Nachrichten entgegen nehmen, zwischenlagern und weitergeben.
3. Vergrößern Sie das zweite Netz wiederum so, dass mit zwei Transitionen und einer Stelle der prinzipielle Systemaufbau modelliert wird. Das Verhalten muss dabei nicht erhalten bleiben!
4. Erweitern Sie das zweite Netz wiederum so, dass das Lager detaillierter modelliert wird. Das Lager soll intern eine Kapazität erhalten, so dass nur bis zu fünf Nachrichten zur Zeit gleichzeitig innerhalb des Lagers befinden können. *Tipp:* Die Anzahl von Marken auf einer Stelle lässt sich nicht direkt beschränken. Überlegen Sie sich also zunächst, wie Sie die Annahme-Transition daran hindern können, zu oft zu schalten und damit mehr als 5 Nachrichten ins Lager zu übernehmen.
5. Fügen Sie einen weiteren Sender hinzu, der über das gleiche Lager Nachrichten verschickt. Achten sie darauf, dass die Kapazität des Lagers erhalten bleibt.
6. Modifizieren Sie das Modell so, dass immer nur zwei Nachrichten auf einmal aus dem Lager entnommen werden.
7. Schalten sie das Netz aus Aufgabenteil 4 oder 5 in einen Zustand, in dem keine weiteren Marken ins Lager gelegt werden können.

Anmerkung: Schicken Sie die Netzzeichnungen der Aufgabe (weiße Fenster, für Aufgabenteile 1 bis 4) als `.rnw` und `.pdf`-Dateien an Ihre Übungsgruppenleitung. Beide Formate werden benötigt. Für die Erstellung der pdf beachten Sie bitte die Hinweise zur Modellierung mit RENEW

Schicken Sie Ihre Simulationen (blaue Fenster, für Aufgabenteil 5) als pdf-Datei an Ihre Übungsgruppenleitung, verwenden Sie dazu (Menü File→Print Drawing und speichern Sie die Datei als pdf. Versuchen Sie nicht die Simulation als `.draw`-Datei abzuspeichern, da das zu Dateiverlust führen kann. Hier reicht die pdf-Version, da die rnw-Versionen in den vorherigen Aufgabenteilen) schon enthalten ist.

Verwenden Sie informative Dateinamen wie:

`FGI_GTT-SS-RRRR_A6_Netz/SimuA_MeierMuellerSchulze.pdf`, wobei

*TT* für Mo oder Di steht, *SS* für die Stunde in der die Übungsgruppe beginnt steht und *RRRR* die Raumnummer angibt,

A6 die Übungsaufgabe A6 bezeichnet,

*Netz/SimuA* zur Kennzeichnung der verschiedenen Netz- und Simulationsvarianten stehen soll und die Nachnamen aller Gruppenmitglieder der Abgabe am Ende stehen soll

Schreiben Sie außerdem in *jedes* Ihrer Modelle alle Nachnamen der jeweiligen Gruppe und den Gruppennamen (Tag + Uhrzeit + Raumnummer), so dass diese auch auf einem Ausdruck sichtbar sind. Nutzen Sie dafür das **Text Tool** in der oberen Werkzeugleiste. Ohne diese Information auf dem Ausdruck könnten keine Punkte vergeben werden.

Schicken Sie Ihrem Übungsgruppenleiter eine zip-Datei, die alle anderen Dateien enthält und die sich in einen einzigen Ordner mit allen Dateien darin entpackt. Dabei sollen die Namen der zip-Datei und des entpackten Ordners dem Namensschema von oben entsprechen.

**Übungsaufgabe 6.5:** Erstellen Sie zwei neue Olat-Frage für den aktuellen Lesestoff der 6. Woche entsprechend den bisherigen Anforderungen. Mindestens eine der Aufgaben soll Fragen zu RENEW enthalten.

von
2

## Hinweise zur Modellierung mit RENEW

- Verwenden Sie die RENEW-Version, auf die schon in den letzten Aufgabenzetteln verwiesen wurde.

Hinweis: RENEW ist im Unix-Cluster des Rechenzentrums vorinstalliert. Es kann von der Kommandozeile aus mit `/local/tgi/renew/bin/unix/renew` gestartet werden. Alternativ steht RENEW unter <http://www.renew.de/> zum Download zur Verfügung.

- Stellen Sie den Formalismus auf P/T-Netze ein (Aufg. 6.4) (Menü **Simulation** → **Formalisms** → **P/T Net Compiler**). Sie müssen diesen Schritt bei jedem Neustart von RENEW wiederholen. Im Netz wird nicht gespeichert, welcher Formalismus verwendet werden soll.
- Erzeugen Sie eine neue Zeichnung mit **File**→**New Net Drawing**
- Erzeugen Sie Transitionen und Plätze mit den Knöpfen der unteren Werkzeugleiste. Die obere Zeile ist im Wesentlichen für das Zeichnen von allgemeinen Graphiken vorgesehen. Diese Graphiken können nicht simuliert werden.
- Ziehen Sie Kanten aus Transitionen und Plätzen heraus, indem Sie im blauen Kringel starten und die Kante bei gedrückter Maustaste zum Zielknoten ziehen. Wenn Sie keinen Zielknoten treffen, wird ein neuer Knoten erzeugt.
- Mit einem rechten Mausklick an einem Platz oder einer Kante erzeugen Sie eine Beschriftung. Ein weiterer rechter Mausklick auf der Beschriftung erlaubt das Bearbeiten. Die Beschriftungen legen Kantengewichte und Anfangsmarkierungen fest. Standardwerte sind Kantengewicht 1 und Anfangsmarkierung 0, wenn keine Beschriftung vorliegt. Bei Verwendung des Formalismus für P/T-Netze werden für die Anschriften natürliche Zahlen erwartet. Die erzeugten Standardanschriften ( $\square$  für Plätze und  $x$  für Kanten) funktionieren mit dem P/T-Formalismus nicht.
- Mit dem Namenswerkzeug (Knopf mit fettem **n**) können Sie Transitionen und Plätze benennen. Namensanschriften sind standardmäßig im Fettdruck gesetzt, so können diese von anderen Anschriften unterschieden werden.
- Testen Sie Ihr Netz, indem Sie eine Simulation starten (Menü **Simulation**→**Simulation Step**). In der Simulationsansicht (blauer Fensterhintergrund) können Sie einzelne Transitionen mit der rechten Maustaste feuern oder per Menübefehl **Simulation Step** einen zufälligen Simulationsschritt ausführen lassen. Mit **Terminate Simulation** wird die Simulation beendet.

Die Schaltfolge sehen Sie mit **Show Simulation Trace**.

- Sie können die Netzzeichnung mit **File** → **Export** → **Current drawing** → **EPS** als skalierbare EPS-Datei exportieren, um sie in Textverarbeitungen einzubinden. Vom PNG-Format ist abzuraten, da dann das Bild mit fester Auflösung verpixelt wird.

*Bisher erreichbare Punktzahl: 72*