

Modellierung und Analyse eingebetteter und verteilter Systeme

Übungsblatt 5

Ausgabe: 12. November, Abgabe: 19. November

Aufgabe 5.1 Einfaches Zustandstransitionssystem (STS) (6 Punkte)

Ein Münzautomat der White Rabbit Ltd. für rote und blaue Pillen soll als STS beschrieben werden:

1. Der Automat besteht aus mehreren Komponenten: Einem Vorratsbehälter für die Pillen, einem Auswurfsschacht für die gekaufte Pille und einem Münzschlitz. Die Aktionen des Kunden veranlassen den Automaten zum Zustandsübergang, deswegen sollte der Kunde selbst auch in die Beschreibung des STS aufgenommen werden.
2. Das Münzzählwerk kann bis zu 3 Münzen gleichzeitig aufnehmen und entleert sich nach jedem Kauf.
3. In den Auswurfsschacht passt nur eine Pille. Diese muss vor Ausgabe der nächsten entleert werden.
4. Eine blaue Pille kostet zwei Münzen, eine rote nur eine Münze.
5. Es gibt zwei Arten von Kunden: neugierige und angepasste. Neugierige Kunden kaufen die rote Pille, angepasste die blaue.
6. Ab und zu muss der Vorratsbehälter aufgefüllt werden, dann sind beide Pillensorten wieder verfügbar.

Die möglichen Aktionen des Automaten sind: *Recharge* (beide Pillensorten werden aufgefüllt), *CustomerArrives*, *CustomerLeaves*, *CoinInsert* (eine Münze wird eingeworfen), *DispenseRed* (Knopf für rote Pille gedrückt), *DispenseBlue* (Knopf für blaue Pille gedrückt) und *TakePill*.

Verwenden Sie die Variablen *stRed* (*bool*, Vorratsbehälter beinhaltet rote Pillen), *stBlue* (*bool*, Vorratsbehälter beinhaltet blaue Pillen), *disp* (Inhalt der Ausgabe), *coins* (Anzahl der Münzen im Zähler), *customer* (Art des Kunden). Der genaue Vorrat an Pillensorten wird nicht mitgezählt.

- a) Geben Sie sinnvolle Wertebereiche für die Variablen an.
- b) Geben Sie die initiale Startverteilung S_0 an.

c) Vervollständigen Sie die obigen Aktionen mit Guards und Effekten.

Aufgabe 5.2 (6 Punkte) CTL Model Checking

Betrachten Sie folgende CTL-Formeln

$$\Phi_1 = EF AG c \text{ sowie } \Phi_2 = A (a U AF c)$$

und das Transitionssystem TS . Entscheiden Sie mit Hilfe des CTL Model Chacking, ob $TS \models \Phi_i$ für $i \in \{1, 2\}$ gilt.

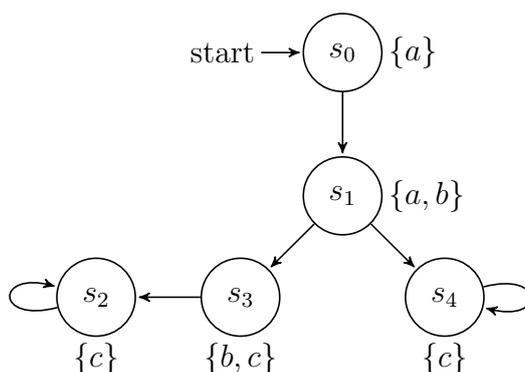


Abbildung 1: Transitionssystem TS

Aufgabe 5.3 (6 Punkte) PROMELA/SPIN Model Checker

Zur Lösung dieser Aufgabe wird das Programm SPIN (s. www.spinroot.com) benötigt. Nutzen Sie die Rechner-Pools der Universität, eine ssh-Verbindung ist dafür ausreichend.

Loggen Sie sich mit Ihrem Pool-Account auf den Rechner marvin.cs.uni-dortmund.de ein und führen Sie folgende Kommandos aus:

```
mkdir spin && cd spin
wget http://spinroot.com/spin/Bin/spin522\_sunos
chmod u+x spin522\_sunos && ln -s spin522\_sunos spin
export PATH=$HOME/spin:$PATH
```

Testen Sie die korrekte Installation durch Ausgabe der Kommandozeilenoptionen: `spin --help`.

Für die Übung sollen zwei Modelle aus [1] ausprobiert werden. Laden Sie bitte dazu das Archiv mit den Beispielen:

```
mkdir examples && cd examples
wget http://spinroot.com/spin/Doc/Book\_extras/examples.tar.gz
gunzip * && tar -xf * && cd App_C
```

Hier folgen PROMELA-Modelle zweier Petri-Netze. Sie können mit dem Kommando `more petrinet1` die erste Datei betrachten.

☞ Wie wird hier das PROMELA-Sprachelement `guard -> effect` zur Beschreibung der Petri-Netze genutzt?

Das erste Modell beinhaltet ein Deadlock. Nutzen Sie SPIN, um mittels Erreichbarkeitsanalyse den Deadlockzustand zu finden: `spin -v petrinet1`.

☞ Warum erkennt SPIN hier die Deadlock-Situation? Welche Suchmethode wendet SPIN hier an?

Das zweite Petri-Netz ist dem Paper unter [2] entnommen. Darin wurde es als frei von Deadlocks bewiesen. Analysieren Sie das Netz mit `spin -v petrinet2`

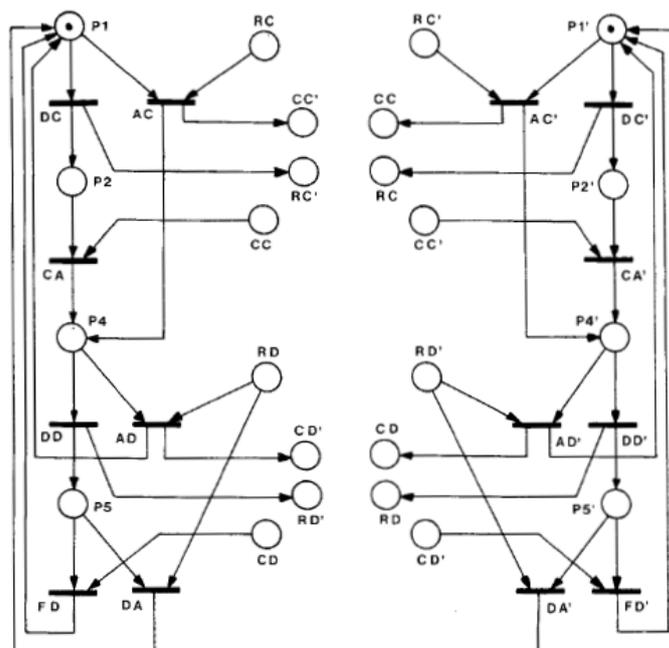


Abbildung 2: Das zweite Petri-Netz

☞ Welcher Fehler tritt hier auf?

Abbildung 2 zeigt das zweite zu untersuchende Petri-Netz. Das Netz stellt einen Kommunikationsablauf zwischen zwei Maschinen dar. Zur besseren Übersicht sind diese in der Zeichnung getrennt. Stellen mit den selben Namen lassen sich zusammenfassen.

Hinweis: Bei Problemen mit dem Pool-Account oder SPIN bitte rechtzeitig Fragen stellen, wir überlegen dann gemeinsam eine Lösung.

Literatur

- [1] Holzmann, G., *The SPIN Model Checker*, Addison-Wesley, 2003.
- [2] Berthelot, G., Terrat, R., "Petri Nets Theory for the Correctness of Protocols," *IEEE Transactions on Communications*, Vol.30, pp.2497-2505, December 1982.