WS 2015/2016 09.02.2014

Übungen zur Vorlesung
Bäume, Ordnungen und Anwendungen
Blatt 14

Prof. Dr. Roland Meyer Keine Abgabe.

Dieses Übungsblatt ist freiwillig. Es gibt keine Abgabe und keine dazugehörige Übung.

Aufgabe 14.1 (Größte Bisimulationsrelation)

a) Seien $K = (AP, S, S_0, \to, \ell)$ und $K' = (AP, S', S'_0, \to', \ell')$ Kripkestrukturen. Zeigen Sie folgende Aussage:

$$K \approx K' \qquad \text{gdw.} \qquad \bigcup_{\substack{R \subseteq S \times S' \text{ ist} \\ \text{Bisimulation}}} R \text{ die Startzustände verbindet.}$$

b) Betrachten Sie die beiden folgenden Kripkestrukturen.

Berechnen Sie die größte Bisimulationsrelation zwischen K und K' mittels Fixpunktiteration von $f_\approx$.

Aufgabe 14.2 (Minimierung von Kripkestrukturen)

a) Gegeben sei eine Kripkestruktur $K = (AP, S, S_0, \to, \ell)$. Bestimmen Sie eine minimale Kripkestruktur $K' = (AP, S', S'_0, \to', \ell')$, so dass $K \approx K'$ gilt.

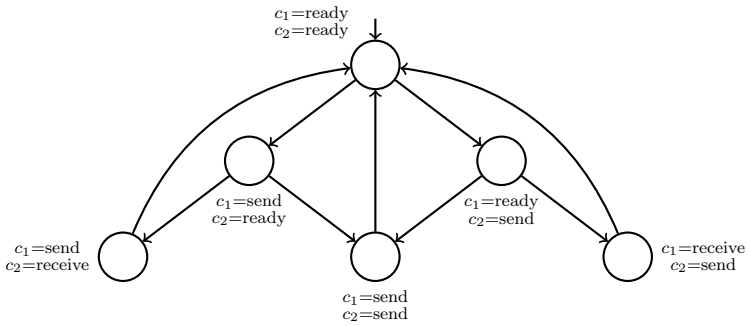
Hinweis: Betrachten Sie die Bisimulation von K mit sich selbst. Zeigen Sie $K \approx K'$, die Minimalität von K' müssen Sie nicht beweisen.

b) Wenden Sie ihr Verfahren aus Aufgabenteil a) an, um die Kripkestruktur K aus Aufgabe 1 zu minimieren.

Aufgabe 14.3 (CTL-Model-Checking)

a) Vervollständigen Sie den Model-Checking-Algorithmus aus der Vorlesung. D.h. geben Sie an, wie man Check($EG\varphi$) berechnet.

b) Betrachten Sie die folgende Kripke-Struktur, die ein einfaches Kommunikationsprotokoll implementiert. Es gibt zwei Clients, die jeder entweder eine Nachricht senden oder eine Nachricht empfangen können. Wer eine Nachricht gesendet hat, wartet darauf, dass sie ankommt, und geht wieder in den Startzustand.



Überprüfen Sie diese Struktur mit dem Verfahren aus der Vorlesung darauf, ob beide Clients beliebig oft senden können.

Aufgabe 14.4 (Abstraktionsverfeinerung)
Betrachten Sie das folgende Programm:

$[x := a]^1$
$[y := b]^2$
while $[x \neq 0]^3$ **do**
    $[x := x - 1]^4$
    $[y := y - 1]^5$
if $[a = b \wedge y \neq 0]^6$ **then**
    $[\text{skip}]^7$
else
    $[\text{skip}]^8$

Wir wollen zeigen, dass man mittels der Abstraktionsverfeinerung aus der Vorlesung nicht beweisen kann, dass Block 7 nicht erreichbar ist. Genauer wollen wir zeigen, dass die Verfeinerung nicht terminiert.

a) Wir betrachten zunächst abstrakte Läufe des Programms. Geben Sie Prädikate $p_i^{(k)}$ für $k \geq 1$ und $i \in \{3, 4, 5\}$ an, die jeweils gelten, wenn Block i zum k-ten Mal betreten wird. *Achtung:* Der Fall $k = 1$ ist ein Sonderfall.

b) Zeigen Sie dann, dass im n-ten Verfeinerungsschritt ein (spurious) Gegenbeispiel mit n Schleifendurchläufen erzeugt wird, aber keines mit weniger als n Durchläufen.

c) Schließen Sie, dass die Verfeinerung nicht terminiert.

Freiwilliges Übungsblatt – keine Abgabe – keine Übung.