FGI-2 – Formale Grundlagen der Informatik II

Modellierung und Analyse von Informatiksystemen

Präsenzlösung 5 : CTL und CTL-Model-Checking

Präsenzteil am 09./10. 11. – Abgabe am 16./17. 11. 2015

Präsenzaufgabe 5.1:

1. Betrachten Sie die Kripke-Strukturen M_1 und M_2 im Skript, Seite 58. Gibt es LTL-Formeln, die die beiden Strukturen unterscheiden? Falls ja, geben Sie welche an!

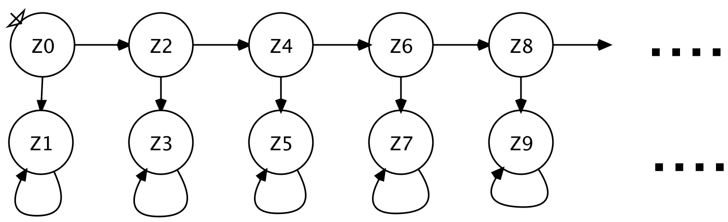
   Lösung: Nein, da jede LTL-Formel in allen Pfaden ab Startzustand gelten muss, um erfüllt zu sein. Der Zeitpunkt der Verzweigung ist in den Pfaden nicht sichtbar, beide Strukturen haben dieselbe Menge möglicher Pfade.

2. M_1 und M_2 wie zuvor: Gibt es CTL-Formeln, die die beiden Strukturen unterscheiden?

   Lösung: Ja, z.B ist die Formel $\mathbf{AXEX}q$ im Startzustand der Struktur M_1 erfüllt, da auf allen möglichen Pfaden im Folgezustand rechtsseitig ein Pfad beginnt, in dessen nächsten Zustand q gilt.

   In M_2 ist die Formel aber nicht erfüllt, da im linksseitigen Pfad im Folgezustand kein solcher Pfad beginnt.

3. Betrachte die folgende Kripkestruktur mit unendlicher Zustandsmenge S, wobei die Zustandsettikettenfunktion für alle $n \in \mathbb{N}$ durch $E_S(z_{2n}) = \emptyset$ und $E_S(z_{2n+1}) = \{p\}$ definiert sei.



   (a) Gilt $f_1 = \mathbf{EF}p$?
   (b) Gilt $f_2 = \mathbf{AGEF}p$?
   (c) Gilt $f_3 = \mathbf{AF}p$?

   Lösung:

   (a) $f_1 = \mathbf{EF}p$ gilt, denn $\pi = z_0 z_1 z_1 \cdots$ ist eine Abwicklung, die $\mathbf{F}p$ erfüllt.
   (b) $f_2 = \mathbf{AGEF}p$ gilt.
   Wir zeigen zunächst, dass in jedem Zustand $\mathbf{EF}p$ gilt.
      i. Zu jedem Zustand z_{2n} existiert der hier startende Pfad $z_{2n} z_{2n+1} z_{2n+1} \cdots$, für den irgendwann (nämlich im 2. Zustand) p gilt.
      ii. Zu jedem Zustand z_{2n+1}, existiert der hier startende Pfad $z_{2n+1} z_{2n+1} \cdots$, für den irgendwann (nämlich sofort) p gilt.

Also $M, z \models \mathbf{EF}p$ für alle Zustände z.

Da es für alle Zustände gilt, folgt dass für jeden aus dem Startzustand startenden Pfad π auch $\mathbf{GEF}p$ gilt.

Da es für jeden Pfad gilt, folgt dass auch $f_2 = \mathbf{AGEF}p$ gilt.

(c) $f_3 = \mathbf{AF}p$ gilt nicht, denn p gilt nirgends auf dem Pfad $z_0 z_2 z_4 z_6 \cdots$

Präsenzaufgabe 5.2: Äquivalenzen in CTL.

1. Formulieren Sie die folgenden Äquivalenzen in natürlicher Sprache und begründen Sie deren Gültigkeit: (i) $\neg\mathbf{G}f \equiv \mathbf{F}(\neg f)$, (ii) $\mathbf{F}f \equiv \mathit{True}\mathbf{U}f$, (iii) $\mathbf{A}f \equiv \neg(\mathbf{E}\neg f)$ und (iv) $\neg\mathbf{X}f \equiv \mathbf{X}\neg f$.

   Lösung:

   (i) $\neg\mathbf{G}f \equiv \mathbf{F}(\neg f)$: Gilt auf einem Pfad nicht immer f, so gibt es einen Zustand, in dem $\neg f$ gilt (und umgekehrt).

   (ii) $\mathbf{F}f \equiv \mathit{True}\mathbf{U}f$: Gilt auf einem Pfad in mindestens einem Zustand f, so gilt auch auf allen Zuständen vor diesem Zustand (trivialerweise) True, also gilt auf diesem Pfad True bis f gilt. Umgekehrt: Wenn True immer gilt, bis mindestens einmal f gilt, so muss es einen Zustand geben, in dem f gilt.

   (iii) $\mathbf{A}f \equiv \neg(\mathbf{E}\neg f)$: Gilt f in allen vom aktuellen Zustand startenden Pfaden, dann gibt es keinen Pfad, auf dem $\neg f$ gelten würde (und umgekehrt).

   (iv) $\neg\mathbf{X}f \equiv \mathbf{X}\neg f$: Wenn es *nicht* stimmt, dass im nächsten Zustand eines Pfades f erfüllt ist, so muss in genau diesem nächsten Zustand zwingend $\neg f$ erfüllt sein. Umgekehrt: Ist im nächsten Zustand $\neg f$ erfüllt, so kann in genau diesem Zustand zwingend f nicht gelten, also gilt *nicht* im nächsten Zustand f. (Das kursive *nicht* steht für das Negationszeichen auf der linken Seite der Äquivalenz).

2. Beweisen Sie die Äquivalenzen:

$$\begin{array}{rcl} \mathbf{AX}g & \equiv & \neg\mathbf{EX}(\neg g) \\ \mathbf{EF}g & \equiv & \mathbf{E}[\mathit{True}\mathbf{U}g] \\ \mathbf{AG}g & \equiv & \neg\mathbf{EF}(\neg g) \\ \mathbf{AF}g & \equiv & \neg\mathbf{EG}(\neg g) \end{array}$$

   Tipp: Nutzen Sie in der Argumentation die einfacheren Äquivalenzen der ersten Teilaufgabe.

   Lösung:

   (a) Mit (iii) und (iv) gilt $\mathbf{AX}g \equiv \neg\mathbf{E}(\neg\mathbf{X}g) \equiv \neg\mathbf{EX}(\neg g)$.

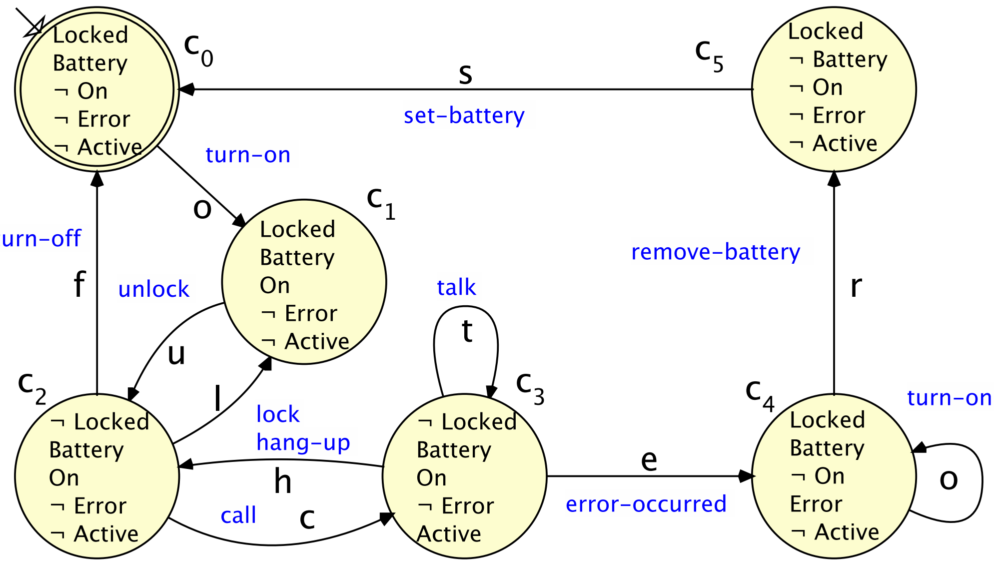
   (b) Mit (ii) gilt $\mathbf{EF}g \equiv \mathbf{E}[\mathit{True}\mathbf{U}g]$.

   (c) Mit (iii) und (i) gilt $\mathbf{AG}g \equiv \neg\mathbf{E}(\neg\mathbf{G}g) \equiv \neg\mathbf{EF}(\neg g)$.

   (d) Mit (iii), (i) und $\neg\neg f \equiv f$ gilt $\mathbf{AF}g \equiv \neg\mathbf{E}\neg\mathbf{F}g \equiv \neg\mathbf{E}\neg\mathbf{F}\neg\neg g \equiv \neg\mathbf{E}\neg\neg\mathbf{G}\neg g \equiv \neg\mathbf{EG}(\neg g)$.

von
6

Übungsaufgabe 5.3: Betrachten Sie das Modell des TS_{cell} Telefons aus dem Aufgabenblatt 4.



TS_{cell}

1. Wenden Sie den CTL-Algorithmus aus dem Skript Abschnitt 4.2 auf folgende Formeln f und g an. Dazu müssen diese natürlich zunächst durch die entsprechenden Äquivalenz-Umformungen umgeformt werden. Geben Sie also zunächt die umgeformten Formeln an auf die der Algorithmus durchgeführt werden kann:

   (a)
   $$f = \Big(\mathbf{AG}(\mathit{Error} \implies \mathbf{E}(\mathit{Error}\,\mathbf{U}\neg \mathit{Battery}))\Big) \wedge \Big(\neg\mathbf{AF}\,\mathit{Active}\Big)$$

   (b)
   $$g = \Big(\mathbf{AGEF}(\mathit{Active})\Big)$$

   Damit man die einzelnen Schritte besser nachvollziehen kann, soll (fast) jeder Rekursionsabschnitt in eine eigene Kopie der Kripkestruktur eingezeichnet werden. Nutzen Sie hierfür die acht Kopien der Abbildung.

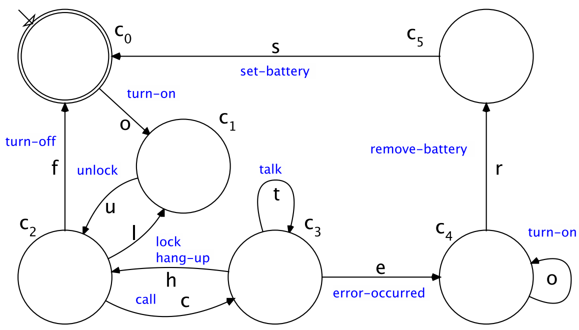
   Notieren Sie an der Teilabbildung jeweils die Teilformel, die der Algorithmus bearbeitet. Geben Sie – wenn nötig – die SZKs o.ä. an.

2. Geben sie eine natürlich-sprachliche Beschreibung für die Bedeutung der Formeln an.

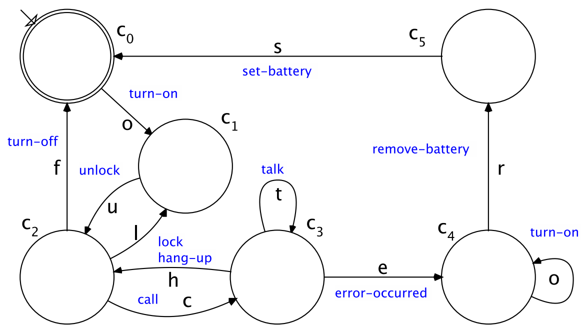
3. Bestimmen Sie $Sat(\phi) := \{s \in S \mid M, s \models \phi\}$ für $\phi \in \{f, g\}$.

4. Entscheiden Sie, ob $M, c_0 \models f$ gilt.

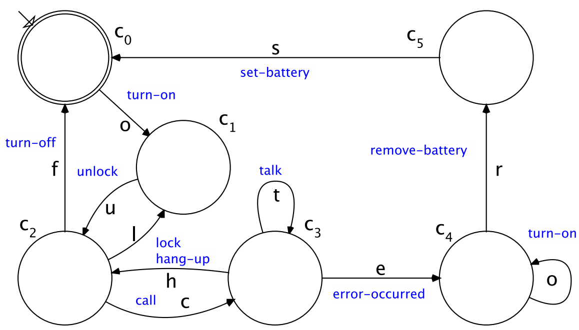
(Teil-)Formel: =



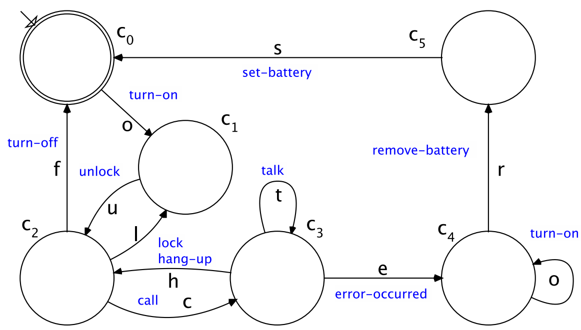
(Teil-)Formel: =



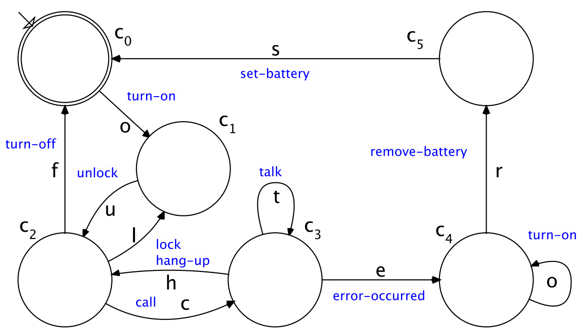
(Teil-)Formel: =



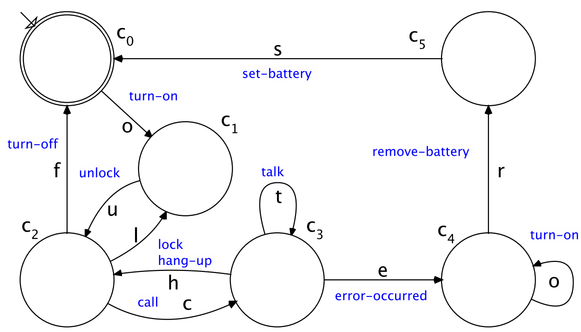
(Teil-)Formel: =



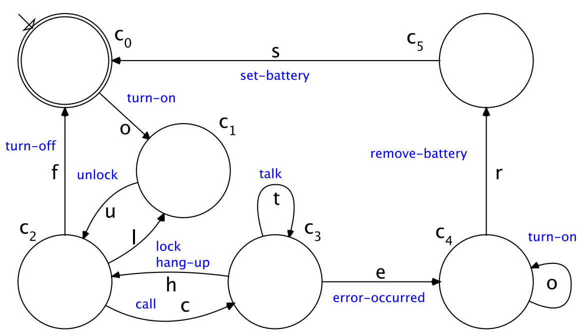
(Teil-)Formel: =



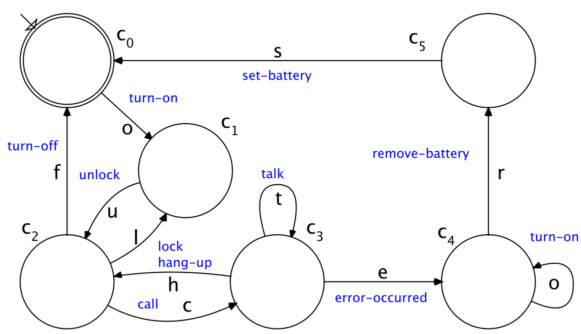
(Teil-)Formel: =



(Teil-)Formel: =



(Teil-)Formel: =



von
6

Übungsaufgabe 5.4: LTL Model-Checking: Führen Sie anhand des folgenden Beispiels den LTL Algorithmus aus.

1. Modellieren Sie zwei Ampeln an einer Kreuzung als Transitionssystem A_i mit $i \in \{1, 2\}$. Beide Ampeln haben vier Phasen: Rot (r_i), Rot-Gelb (rg_i), Grün (gr_i), Gelb (g_i) . Ampel 1 startet in der Phase Rot, während Ampel 2 mit der Phase Grün beginnt. Annotieren Sie die Zustände mit den Etikettenanschriften.

2. Konstruieren sie aus den beiden Transitionssystemen (A_1 und A_2) das Produkttransitionssystem $A_1 \otimes A_2$ mit einer für zwei Ampeln geeigneten Synchronisationsrelation.

3. Eine geeignete Spezifikation (der ersten Bedingung) für das Gesamtsystem wäre durch die Formel $\phi = G\neg(gr_1 \wedge gr_2)$ gegeben. Sei M_ϕ der Büchi-Automat, der die Sprache $L^\omega(\phi)$ akzeptiert. Zu untersuchen ist nun, ob $L^\omega(A_1 \otimes A_2) \subseteq L^\omega(\phi) = L^\omega(M_\phi)$ stimmt. Dies entspräche der Frage, ob $L^\omega(A_1 \otimes A_2) \cap \overline{L^\omega(M_\phi)} = \emptyset$ korrekt ist. Einfacher ist es $L^\omega(A_1 \otimes A_2) \cap L^\omega(M_{\neg\phi}) = \emptyset$ zu prüfen.

   Erstellen Sie also die Formel $\neg\phi$ und daraus dann einen Büchiautomat $M_{\neg\phi}$, der dieselbe Sprache akzeptiert. (Zwei Zustände reichen aus.)

4. Erstellen Sie zunächst aus dem etikettierten Transitionssystem ($A_1 \otimes A_2$) einen Büchiautomaten B, der die Etikettensprache von ($A_1 \otimes A_2$) akzeptiert.

5. Erstellen Sie den Produktautomat $B \otimes M_{\neg\phi}$.

6. Erfüllt das System die Spezifikation ϕ?

7. *(Bonusaufgabe)* Führen Sie die Prüfung der zweiten Bedingung $\psi = GFgr_1$ exemplarisch für eine der beiden Ampeln in gleicher Weise durch.

Hinweis: Verwenden Sie bitte abkürzende Schreibweisen für die Etiketten. So kann beispielsweise bei einem Etikettenalphabet $AP = \{a, b, c\}$ anstelle von $\{\emptyset, \{b\}, \{c\}, \{b, c\}\}$ einfach $\neg a$ geschrieben werden.

von
1

Bonusaufgabe 5.5: Erstellen Sie eine neue Olat-Frage für den aktuellen Lesestoff entsprechend den bisherigen Anforderungen.

Bisher erreichbare Punktzahl: 60