FGI-2 – Formale Grundlagen der Informatik II

Modellierung und Analyse von Informatiksystemen

Musterlösung 5 : CTL und CTL-Model-Checking

Präsenzteil am 11./12. 11. – Abgabe am 18./19. 11. 2013

Präsenzaufgabe 5.1:

1. Betrachten Sie die Kripke-Strukturen M_1 und M_2 im Skript, Seite 50. Gibt es LTL-Formeln, die die beiden Strukturen unterscheiden? Falls ja, geben Sie welche an!

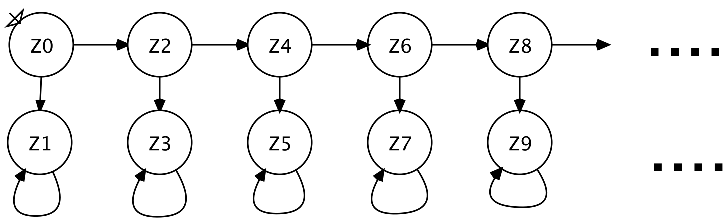
   Lösung: Nein, da jede LTL-Formel in allen Pfaden ab Startzustand gelten muss, um erfüllt zu sein. Der Zeitpunkt der Verzweigung ist in den Pfaden nicht sichtbar, beide Strukturen haben dieselbe Menge möglicher Pfade.

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:

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

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

   Lösung:

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

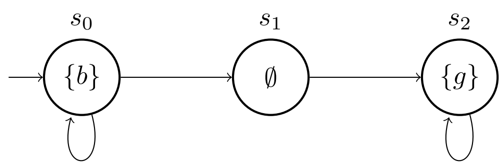
   (b) Mit (ii) gilt $\mathbf{EF}g \equiv \mathbf{E}[\mathit{True}\mathbf{U}g]$.

   (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 $\mathbf{AF}g \equiv \neg\mathbf{E}\neg\mathbf{F}g \equiv \neg\mathbf{E}\neg\mathbf{F}\neg\neg g \equiv \neg\mathbf{E}\neg\neg\mathbf{G}\neg g \equiv \neg\mathbf{EG}(\neg g)$.

Übungsaufgabe 5.3: Betrachten Sie das Kripke-Modell M_{AKW} eines Atomkraftwerkes. In dem Normalbetrieb (Zustand s_0: „Betrieb" b) kann eine Störung (Zustand s_1) auftreten, wonach der Störbetrieb (Zustand s_2: „gestört" g) aufgenommen wird.

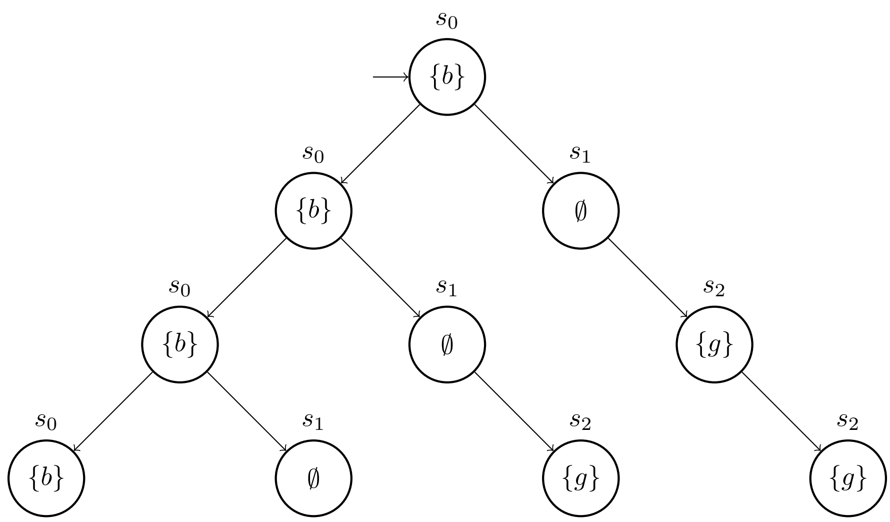
von 6



M_{AKW}

1. Konstruieren Sie den Abwicklungsbaum $Abwicklung_{AKW}$ (siehe Skript Abb. 3.2) bis zur Tiefe 4 und bezeichnen dazu die Zustände und Etikette wie in M_{AKW}.

   Lösung: $Abwicklung_{AKW}$



2. Wieder sei $Sat(\alpha)$ die Menge aller Zustände, die α erfüllen. Bestimmen Sie mit Hilfe von $Abwicklung_{AKW}$ (oder direkt mit M_{AKW}) die Mengen

   (a) $Sat(\alpha_1)$ mit $\alpha_1 = \mathbf{EX}b$,
   (b) $Sat(\mathbf{AG}\alpha_1)$,
   (c) $Sat(\alpha_2)$ mit $\alpha_2 = \mathbf{AG}\neg b$ und
   (d) $Sat(\mathbf{EX}\alpha_2)$.

   Lösung:

   (a) $Sat(\alpha_1) = \{s_0\}$,
   (b) $Sat(\mathbf{AG}\alpha_1) = \emptyset$,
   (c) $Sat(\alpha_2) = \{s_1, s_2\}$ und
   (d) $Sat(\mathbf{EX}\alpha_2) = \{s_0, s_1, s_2\}$.

3. Interpretieren Sie für M_{AKW} die Formeln $\beta_1 = \mathbf{AGEX}b$ und $\beta_2 = \mathbf{EXAG}\neg b$ und entscheiden Sie (unter zu Hilfenahme von 2.), ob sie für M_{AKW} gelten, d.h. ob

   (a) $M_{AKW} \models \mathbf{AGEX}b$ und
   (b) $M_{AKW} \models \mathbf{EXAG}\neg b$

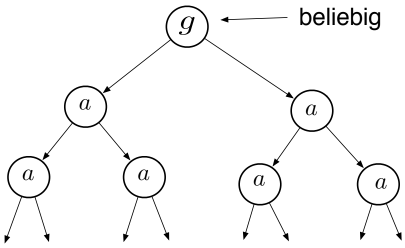
   gelten. Dabei sei (analog zu Def. 3.5) $M \models \alpha :\Leftrightarrow \forall s \in S_0 : M, s \models \alpha$.

   Lösung:

   (a) β_1 : Egal, was passiert, das System kann im nächsten Schritt wieder in Betrieb sein. β_1 gilt nicht, da $s_0 \notin Sat(\beta_1)$. (vergl. 2).
   (b) β_2 : Es gibt einen ersten Schritt, wonach das System nicht mehr in Betrieb ist und auch später nie wieder in Betrieb geht. β_2 gilt, da $s_0 \in Sat(\beta_2)$ (vergl. 2.).

4. (a) Beweisen Sie für alle Aussagen a: $\mathbf{AXAG}a \equiv \mathbf{AGAX}a$.
   Hinweis: Konstruieren Sie (symbolisch) für beide Seiten den Abwicklungsbaum.
   (Anmerkung: Die Äquivalenz $\equiv$ ist definiert als: $f \equiv g$ gilt genau dann, wenn für jedes Modell M gilt: $M \models f$ gdw. $M \models g$. f ist also in jedem Modell wahr, in dem auch g wahr ist und andersherum. (Siehe auch die Definition zu Beginn von Abschnitt 3.4 im Skript auf Seite 40.))

Lösung: In beiden Fällen ergibt sich folgende Abwicklung:



(b) Beweisen Sie, dass folgende Äquivalenz **nicht** gilt: $\mathbf{EXEG}(\neg b \wedge \neg g) \equiv \mathbf{EGEX}(\neg b \wedge \neg g)$.
Hinweis: Benutzen Sie M_{AKW} zur Konstruktion eines Gegenbeispiels.

Lösung: In M_{AKW} gilt $Sat(\mathbf{EG}(\neg b \wedge \neg g)) = \emptyset$ und daher auch $Sat(\mathbf{EXEG}(\neg b \wedge \neg g)) = \emptyset$. Die rechte Seite wird jedoch durch die Zustandsfolge $\pi = s_0 s_0 s_0 \cdots$ erfüllt.

5. Indem man alle Symbole **A** und **E** streicht, erhält man aus einer CTL-Formel eine LTL-Formel. Ist die so erhaltene Formel äquivalent zur ursprünglichen (im Sinne der Definition vor Satz 3.14 auf Seite 40)?

(a) Beweisen Sie als positives Beispiel: $\mathbf{AGAX}b \equiv \mathbf{GX}b$.
(Hinweis: Die Äquivalenz ist hier so zu verstehen, dass jedes Modell, dass die CTL-Formel $\mathbf{AGAG}x$ auch die LTL-Formel $\mathbf{GX}b$ erfüllt und umgekehrt.)

Lösung: In beiden Fällen gilt ab dem zweiten Zustand jeden Pfades immer b.

(b) Beweisen Sie als negatives Beispiel: $\mathbf{EG}b \equiv \mathbf{G}b$.
Hinweis: Benutzen Sie M_{AKW} zur Konstruktion eines Gegenbeispiels.

Lösung: In der Kripkestruktur M_{AKW} gibt es zwei unendliche Rechnungen: $\pi_1 = s_0^\omega$ und $\pi_2 = s_0 s_1 s_2^\omega$ mit den zugehörigen Zustandsetikettenfolgen $E_S(\pi_1) = \{b\}^\omega$ und $E_S(\pi_2) = \{b\}\emptyset\{g\}^\omega$ (Def. 2.18).

i. $E_S(\pi_1)$ erfüllt $\mathbf{EG}b$ als CTL-Formel. Also gilt: $M_{AKW} \models_{CTL} \mathbf{EG}b$.
ii. $E_S(\pi_1)$ erfüllt $\mathbf{G}b$, aber $E_S(\pi_2)$ erfüllt $\mathbf{G}b$ nicht. Daher gilt: $M_{AKW} \not\models_{LTL} \mathbf{G}b$.

Übungsaufgabe 5.4: Wenden Sie den CTL-Model-Checking-Algorithmus (Skript Abschnitt 4.2) auf die beiden CTL-Formeln $\beta_1 = \mathbf{AGEX}b$ und $\beta_2 = \mathbf{EXAG}\neg b$ von Aufgabe 5.3.3 und die Kripke-Struktur M_{AKW} an. Gehen Sie dabei fogendermaßen vor:

von
6

1. Bringen Sie β_1 und β_2 in eine Form β_1' und β_2', die nur **EX**, **EG** und **EU** verwendet. Auch **EF** kann sinnvollerweise als spezielle Form von **EU** benutzt werden.

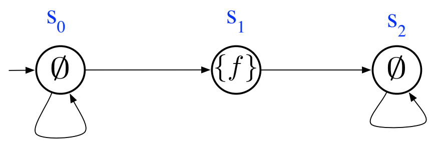
   Lösung: $\beta_1 = \mathbf{AGEX}b \equiv \mathbf{AG}\neg\neg\mathbf{EX}b \equiv \neg\mathbf{EF}\neg\mathbf{EX}b = \beta_1'$ $(\equiv \neg\mathbf{E}[true\mathbf{U}\neg\mathbf{EX}b])$
   und $\beta_2 = \mathbf{EXAG}\neg b \equiv \mathbf{EX}\neg\neg\mathbf{AG}\neg b \equiv \mathbf{EX}\neg\mathbf{EF}\neg\neg b \equiv \mathbf{EX}\neg\mathbf{EF}b = \beta_2'$ $(\equiv \mathbf{EX}\neg\mathbf{E}[true\mathbf{U}b])$

2. Wenden Sie den CTL-Algorithmus nicht auf den Graphen von M_{AKW} an, sondern in Form folgender Tabelle:

Teilformel	Zustand s_0	Zustand s_1	Zustand s_2
b	$+$	$-$	$-$
$\mathbf{EX}b$	$\cdots$	$\cdots$	$\cdots$
$\cdots$	$\cdots$	$\cdots$	$\cdots$

In der linken Spalte steht die zu prüfende Formel und aufsteigend alle Teilformeln, beginnend mit der kleinsten Teilformel b. Unter „Zustand s_i“ steht ein $+$, wenn die Teilformel der Zeile im entsprechenden Schritt im Graphen an diesen Zustand zu schreiben ist. Im anderen Fall steht ein $-$.

Lösung:



Teilformel	Zustand s_0	Zustand s_1	Zustand s_2
b	$+$	$-$	$-$
$\mathbf{EX}b$	$+$	$-$	$-$
$\neg\mathbf{EX}b$	$-$	$+$	$+$
$\mathbf{EF}\neg\mathbf{EX}b$	$+$	$+$	$+$
$\beta_1' = \neg\mathbf{EF}\neg\mathbf{EX}b$	$-$	$-$	$-$

Teilformel	Zustand s_0	Zustand s_1	Zustand s_2
b	$+$	$-$	$-$
$\mathbf{EF}b$	$+$	$-$	$-$
$\neg\mathbf{EF}b$	$-$	$+$	$+$
$\beta_2' = \mathbf{EX}\neg\mathbf{EF}b$	$+$	$+$	$+$

3. Entscheiden Sie, ob $M_{AKW} \models \beta_1$ und $M_{AKW} \models \beta_2$ gilt. Vergleichen Sie die letzten Zeilen der Tabellen mit Ihrem Ergebnis zu $Sat(\beta_1)$ und $Sat(\beta_2)$ aus Aufgabe 5.3.2. und erklären Sie Übereinstimmungen.

   Lösung: Es gilt nicht $M_{AKW} \models \beta_1$, da der Anfangszustand s_0 für β_1' mit $-$ markiert ist. Es gilt $M_{AKW} \models \beta_2$, da der Anfangszustand s_0 für β_2' mit $+$ markiert ist. Auch für die Zustände s_1 und s_2 gibt die letzte Zeile an, ob die jeweilige Formel dort gilt.

4. Anmerkung: Zu dem Verfahren der Umwandlung einer CTL-Formel in eine LTL-Formel gibt es ein interessantes Theorem. Wenn die durch Streichen der Quantoren **A** und **E** erzeugte LTL-Formel nicht äquivalent zur ursprünglichen ist, dann gibt es überhaupt keine LTL-Formel, die das leistet!

Bisher erreichbare Punktzahl: 60