

Temporale Logik und Zustandssysteme Lösungsvorschlag

Aufgabe 4-1

Herleitungen in Σ_{LTL}

(keine Abgabe)

Leiten Sie folgende Regel und Formel in Σ_{LTL} her. Verwenden Sie dazu nur die Axiome und Regeln von Σ_{LTL} sowie (prop).

a) $\Box A \rightarrow B \vdash \Box A \rightarrow \Box B$

Lösung:

- | | | |
|-----|----------------------------------|--------------|
| (1) | $\Box A \rightarrow B$ | (Ann) |
| (2) | $\Box A \rightarrow \Box \Box A$ | (ltl3)(prop) |
| (3) | $\Box A \rightarrow \Box B$ | (ind)(1)(2) |

b) $(\Box A \wedge \Box B) \rightarrow \Box(A \wedge B)$ (eine Richtung von (T15))

Lösung: Man beachte, dass $A \wedge B$ die Abkürzung der Formel $\neg(A \rightarrow \neg B)$ ist. Demnach muss man folgende Formel herleiten: $\neg(\Box A \rightarrow \neg \Box B) \rightarrow \Box \neg(A \rightarrow \neg B)$. Eine mögliche Herleitung dieser Formel ist die folgende:

- | | | |
|-----|-----------------------------------------------------------------------------------------|--------------------|
| (1) | $\neg \Box(A \rightarrow \neg B) \rightarrow \Box \neg(A \rightarrow \neg B)$ | (ltl1),(prop) |
| (2) | $\neg \Box \neg(A \rightarrow \neg B) \rightarrow \neg \neg \Box(A \rightarrow \neg B)$ | (prop),(1) |
| (3) | $\neg \Box \neg(A \rightarrow \neg B) \rightarrow \Box(A \rightarrow \neg B)$ | (prop),(2) |
| (4) | $\Box(A \rightarrow \neg B) \rightarrow (\Box A \rightarrow \Box \neg B)$ | (ltl2) |
| (5) | $(\Box A \rightarrow \Box \neg B) \rightarrow (\Box A \rightarrow \neg \Box B)$ | (ltl1),(prop) |
| (6) | $\neg \Box \neg(A \rightarrow \neg B) \rightarrow (\Box A \rightarrow \neg \Box B)$ | (3),(4),(5),(prop) |
| (7) | $\neg(\Box A \rightarrow \neg \Box B) \rightarrow \Box \neg(A \rightarrow \neg B)$ | (prop),(6) |

Aufgabe 4-2

Herleitungen in Σ_{LTL}

(6+4 Punkte)

Leiten Sie folgende Formeln und Regel in Σ_{LTL} her. Verwenden Sie dazu nur die Axiome und Regeln von Σ_{LTL} sowie (prop) und gegebenenfalls selbst hergeleitete Sätze und Regeln.

a) $\Box A \leftrightarrow \Box \Box A$ (T10)

Lösung:

- | | | |
|-----|--------------------------------------|--------------|
| (1) | $\Box \Box A \rightarrow \Box A$ | (ltl3)(prop) |
| (2) | $\Box A \rightarrow \Box A$ | (taut) |
| (3) | $\Box A \rightarrow \Box \Box A$ | (ltl3)(prop) |
| (4) | $\Box A \rightarrow \Box \Box A$ | (ind)(2)(3) |
| (5) | $\Box A \leftrightarrow \Box \Box A$ | (prop)(1)(4) |

b) $\Box(A \wedge B) \rightarrow \Box A \wedge \Box B$ (eine Richtung von (T15))

Lösung:

- (1) $A \wedge B \rightarrow A$ (taut)
- (2) $\circ(A \wedge B \rightarrow A)$ (nex)(1)
- (3) $\circ(A \wedge B \rightarrow A) \rightarrow \circ(A \wedge B) \rightarrow \circ A$ (ltl2)
- (4) $\circ(A \wedge B) \rightarrow \circ A$ (mp)(2)(3)
- (5) $A \wedge B \rightarrow B$ (taut)
- ...
- (8) $\circ(A \wedge B) \rightarrow \circ B$ (mp)(6)(7)
- (9) $\circ(A \wedge B) \rightarrow \circ A \wedge \circ B$ (prop)(4)(8)

c) $A \rightarrow \circ B \vdash A \rightarrow \diamond B$ (som)

Lösung: Man beachte, dass $C \rightarrow \diamond D \equiv C \rightarrow \neg \square \neg D \equiv \square \neg C \rightarrow \neg D$ gilt.

- (1) $\square \neg B \rightarrow \neg B$ (ltl3)(prop)
- (2) $\circ(\square \neg B \rightarrow \neg B)$ (nex)(1)
- (3) $\circ \square \neg B \rightarrow \circ \neg B$ (ltl2)(prop)(2)
- (4) $\square \neg B \rightarrow \circ \square \neg B$ (ltl3)(prop)
- (5) $\square \neg B \rightarrow \circ \neg B$ (prop)(3)(4)
- (6) $\square \neg B \rightarrow \neg \circ B$ (ltl1)(prop)(5)
- (7) $\circ B \rightarrow \diamond B$ (prop)(6)
- (8) $A \rightarrow \circ B$ (Ann)
- (9) $A \rightarrow \diamond B$ (prop)(7)(8)

d) $\square(A \rightarrow B) \rightarrow (\square A \rightarrow \square B)$ (T19)

(+4 Sonderpunkte)

Lösung:

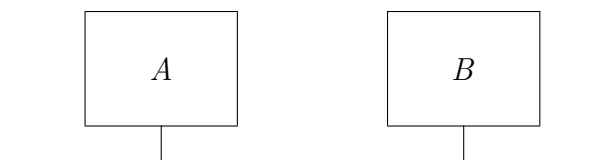
- (1) $(\square(A \rightarrow B) \wedge \square A \rightarrow \square B) \rightarrow (\square(A \rightarrow B) \rightarrow (\square A \rightarrow \square B))$ (taut)
- (2) $\square(A \rightarrow B) \rightarrow (A \rightarrow B)$ (ltl3),(prop)
- (3) $\square A \rightarrow A$ (ltl3),(prop)
- (4) $\square(A \rightarrow B) \wedge \square A \rightarrow ((A \rightarrow B) \wedge A)$ (2),(3),(prop)
- (5) $\square(A \rightarrow B) \wedge \square A \rightarrow B$ (4),(prop)
- (6) $\square(A \rightarrow B) \wedge \square A \rightarrow \circ \square(A \rightarrow B) \wedge \circ \square A$ (ltl3),(prop)
- (7) $\circ \square(A \rightarrow B) \wedge \circ \square A \rightarrow \circ(\square(A \rightarrow B) \wedge \square A)$ (Aufg. 4-1,b))
- (8) $\square(A \rightarrow B) \wedge \square A \rightarrow \circ(\square(A \rightarrow B) \wedge \square A)$ (6),(7),(prop)
- (9) $\square(A \rightarrow B) \wedge \square A \rightarrow \square B$ (ind),(5),(8)
- (10) $\square(A \rightarrow B) \rightarrow (\square A \rightarrow \square B)$ (1),(9),(prop)

Aufgabe 4-3

Wechselseitiger Ausschluß

(6 Punkte)

An einem Hardware-Bus sind zwei Komponenten A und B angeschlossen. Eine Bus-Vorrangschaltung (*bus arbiter*) verhindert den gleichzeitigen Zugriff der Komponenten auf den Bus.



Will Komponente A in einem bestimmten Taktzyklus auf den Bus zugreifen, so setzt sie das *req1*-Bit; erlaubt die Vorrangschaltung den Zugriff, so antwortet sie durch Setzen des *ack1*-Bits im gleichen Taktzyklus. Analog werden die Bits *req2* und *ack2* für die Kommunikation mit der Komponente B benutzt.

Um keine Komponente zu bevorzugen, wird ein *prio*-Bit verwendet, das das Verhalten der Vorrangschaltung beim gleichzeitigen Zugriff von *A* und *B* auf den Bus wie folgt steuert: Ist *prio* gesetzt, so erhält *A* den Vorrang, ansonsten *B*. Der Wert des *prio*-Bits wird anschließend negiert.

Wenn nicht beide Komponenten gleichzeitig auf den Bus zugreifen wollen, bleibt der Wert des *prio*-Bits unverändert.

- a) Modellieren Sie das Verhalten der Schaltung durch eine geeignete Formelmengende \mathcal{A} der temporalen Aussagenlogik LTL. Die Antwort der Schaltung (d.h. das Setzen der *ack*-Bits) erfolgt dabei im selben Taktzyklus wie die Anforderung (ausgedrückt durch die *req*-Bits).

Lösung: Eine mögliche Modellierung der Schaltung ist gegeben durch die Formeln

$$\mathcal{A} = \{ack1 \leftrightarrow req1 \wedge (\neg req2 \vee prio), \\ ack2 \leftrightarrow req2 \wedge (\neg req1 \vee \neg prio), \\ \neg(req1 \wedge req2) \rightarrow (\bigcirc prio \leftrightarrow prio), \\ req1 \wedge req2 \rightarrow (\bigcirc prio \leftrightarrow \neg prio)\}$$

Diese Formeln beschreiben das Verhalten der Schaltung in jedem Zustand. Man könnte daher auch $\Box(ack1 \leftrightarrow req1 \wedge (\neg req2 \vee prio))$ usw. schreiben, dieser äußerste \Box -Operator ist aber nicht notwendig, weil wir uns im Folgenden auf Folgerungen aus diesen Formeln konzentrieren und die Folgerungsbeziehung implizit über alle Zustände quantifiziert.

- b) Beschreiben Sie folgende Eigenschaften durch Formeln von \mathcal{L}_{LTL} und zeigen Sie, dass diese aus \mathcal{A} folgen:

- Die Komponenten *A* und *B* erhalten nicht gleichzeitig das Recht zum Zugriff auf den Bus.
- Signalisiert mindestens eine Komponente den Wunsch zum Zugriff auf den Bus, so erhält mindestens eine Komponente das Recht dazu.
- Will Komponente *A* unendlich oft auf den Bus zugreifen, so erhält sie unendlich oft das Recht dazu.

Lösung: Es sei $K = (\eta_0, \eta_1, \dots)$ eine temporale Struktur mit $\models_{\overline{K}} A$ für alle $A \in \mathcal{A}$.

- $excl \equiv \neg(ack1 \wedge ack2)$:

Es sei $i \in \mathbb{N}_0$ beliebig gewählt, wir müssen zeigen, dass $K_i(excl) = \text{tt}$ gilt. Angenommen nicht, dann folgt aus der Gültigkeit der Formeln in \mathcal{A} , dass $\eta_i(req1) = \eta_i(req2) = \text{tt}$ ist und außerdem $K_i(\neg req2 \vee prio) = K_i(\neg req1 \vee \neg prio) = \text{tt}$ gilt. Wegen $\eta_i(req1) = \eta_i(req2) = \text{tt}$ müßte also $\eta_i(prio) = \text{tt}$ und $\eta_i(prio) = \text{ff}$ gelten, Widerspruch.

- $resp \equiv req1 \vee req2 \rightarrow ack1 \vee ack2$:

Wieder sei $i \in \mathbb{N}_0$ beliebig gewählt, ferner sei $K_i(req1 \vee req2) = \text{tt}$ (ansonsten gilt die Aussage trivialerweise). O.B.d.A. gelte $\eta_i(req1) = \text{tt}$, der andere Fall ist symmetrisch. Wir unterscheiden drei Fälle:

- Ist $\eta_i(req2) = \text{ff}$, so folgt $\eta_i(ack1) = \text{tt}$ unmittelbar aus \mathcal{A} .
- Ist $\eta_i(prio) = \text{tt}$, so folgt ebenfalls $\eta_i(ack1) = \text{tt}$ aus \mathcal{A} .
- Ist schließlich $\eta_i(req2) = \text{tt}$ und $\eta_i(prio) = \text{ff}$, so folgt $\eta_i(ack2) = \text{tt}$, und damit ebenfalls die Behauptung.

- $fair \equiv \Box \diamond req1 \rightarrow \Box \diamond ack1$:

Es sei $i \in \mathbb{N}_0$ beliebig gewählt, und es gelte $K_i(\Box \diamond req1) = \text{tt}$. Wir müssen zeigen, dass $K_i(\Box \diamond ack1) = \text{tt}$ gilt. Angenommen nicht, dann existiert $j \geq i$, so dass $K_j(ack1) = \text{ff}$ für alle $k \geq j$.

Nach Annahme existieren $j \leq m < n$ mit $\eta_m(req1) = \eta_n(req1) = \text{tt}$ und $\eta_l(req1) = \text{ff}$ für alle $m < l < n$. Wäre $\eta_m(req2) = \text{ff}$ oder $\eta_m(prio) = \text{tt}$, so folgte $\eta_m(ack1) = \text{tt}$ im Widerspruch zur Annahme. Also gilt $\eta_m(req2) = \text{tt}$ und $\eta_m(prio) = \text{ff}$, und es folgt $\eta_{m+1}(prio) = \text{tt}$. Wegen $\eta_l(req1) = \text{ff}$ für $m < l < n$ folgt $K_l(req1 \wedge req2) = \text{ff}$ und daher $K_{l+1}(prio) = K_l(prio)$ für $m < l < n$. Also ist $\eta_n(prio) = \text{tt}$. Mit der Annahme $\eta_n(req1) = \text{tt}$ folgt $\eta_n(ack1) = \text{tt}$, und wir haben einen Widerspruch erreicht.

Abgabe: Mittwoch, den 15.11.2006, vor der Übung.