

Temporale Logik und Zustandssysteme

Aufgabe 10-1 Modellierung und Verifikation eines Fahrstuhls (14 Punkte)

Ein Aufzug bedient n Stockwerke eines Hauses. Der Aufzugschacht wird auf jedem Stockwerk durch eine elektrische Schiebetür abgeschlossen. Zur Anforderung des Aufzugs befinden sich auf jedem Stockwerk Druckknöpfe (je einer im untersten bzw. obersten Stockwerk, ansonsten je zwei für "Fahrt nach oben" bzw. "Fahrt nach unten"). In der Aufzugskabine befindet sich eine Leiste von Druckknöpfen (je einer pro Stockwerk) zur Eingabe der Fahrtziele.

- a) Modellieren Sie die Steuerung für den Aufzug und die Türen durch ein faires Zustandssystem Γ mit Startzuständen (frSTS) bezüglich einer geeigneten Signatur SIG und prädikatenlogischen Struktur S . Berücksichtigen Sie dabei die folgenden Anforderungen:
- Wenn keine Fahrtwünsche vorliegen, bleibt der Aufzug stehen.
 - Der Aufzug ändert seine Richtung nur dann, wenn keine weiteren Fahrtwünsche in der bisherigen Richtung vorliegen.
 - Erreicht der Aufzug ein Stockwerk, für das eine Anforderung vorliegt, so hält der Aufzug an. Dies gilt nicht, falls der Fahrtwunsch die entgegengesetzte Richtung angibt, aber weitere Fahrtwünsche für die aktuelle Fahrtrichtung anliegen.
 - Die Fahrt von einem Stockwerk zum nächsten dauert mehrere Taktzyklen der Steuerung und kann daher nicht als (atomare) Aktion modelliert werden. Insbesondere können während der Fahrt neue Fahrtwünsche eintreffen.
 - Der Aufzug darf nur losfahren, wenn die Tür des aktuellen Stockwerks ganz verschlossen ist. Das Öffnen bzw. Schließen einer Tür dauert mehrere Taktzyklen und kann daher nicht als (atomare) Aktion modelliert werden. Während des Schließens einer Tür kann diese durch erneutes Drücken der Anforderungstaste (im Lift oder am Stockwerk) wieder geöffnet werden.

Verwenden Sie dabei die Menge $Act = \{req_up_i, req_dn_i, req_dest_i, start_lift, pass_floor, stop_lift, door_open, close_door, door_closed, reopen_door \mid i \in \{1, \dots, n\}\}$ mit folgenden informellen Bedeutungen:

- req_up_i, req_down_i : Im Stockwerk i wird ein Wunsch nach einer Fahrt nach oben/unten angezeigt.
 - req_dest_i : Im Fahrstuhl wird das Stockwerk i als Fahrtziel angefordert.
 - $start_lift$: Nach dem Schliessen der Türen setzt sich der Lift in Bewegung.
 - $pass_floor$: Der fahrende Lift passiert ein Stockwerk, ohne anzuhalten.
 - $stop_lift$: Der fahrende Lift erreicht ein Stockwerk, in welchem er anhält.
 - $door_open$: Eine sich öffnende Tür erreicht den Zustand, in dem sie ganz geöffnet ist.
 - $close_door$: Das Schliessen einer offenen Tür wird ausgelöst.
 - $door_closed$: Das Schliessen einer Tür wird abgeschlossen.
 - $reopen_door$: Eine geschlossene oder sich schliessende Tür wird wieder geöffnet, sofern das möglich ist.
- b) Geben Sie für jede Aktion von Γ geeignete Ausführbarkeitsbedingungen an und beschreiben Sie die Wirkung der Aktionen durch Formeln von \mathcal{L}_{TLF} .
- c) Beweisen Sie folgende Aussagen für die Aufzugsteuerung:
1. Wenn der Aufzug in Bewegung ist, liegen Fahrtwünsche in der aktuellen Richtung vor.
 2. Der Aufzug hält an einem Stockwerk nur dann an, wenn eine Anforderung für dieses Stockwerk vorliegt.

Aufgabe 10-2**Dining Philosophers**

(keine Abgabe)

Die dinierenden Philosophen sind ein klassisches Modellierungsproblem von E. Dijkstra, gestellt als Klausuraufgabe unter dem Namen „dining quintuple“ und von T. Hoare in „dining philosophers“ umgetauft.

An einem runden Tisch sitzen $n > 1$ Philosophen. Philosophen verbringen ihr Leben fortwährend mit Essen und Denken (was beides nicht zur gleichen Zeit stattfinden kann). Vor jedem Philosophen steht eine Schale Reis; zwischen je zwei Philosophen liegt ein Stäbchen. Um essen zu können, benötigt der Philosoph sowohl sein rechtes als auch sein linkes Stäbchen. Dabei gilt:

- Will ein Philosoph essen, so nimmt er zuerst sein rechtes, dann sein linkes Stäbchen.
 - Wird ein Stäbchen vom jeweiligen Nachbarn benutzt, so wartet der Philosoph, bis der Nachbar das Stäbchen freigibt.
 - Nach dem Essen legt der Philosoph ohne Unterbrechung beide Stäbchen ab und denkt wieder.
 - Ein aufgenommenes Stäbchen legt ein Philosoph nicht wieder zurück, bis er das Essen beendet hat.
 - Essen und Denken kann beliebig lange andauern, das Aufnehmen und Ablegen der Stäbchen hingegen wird schnellstmöglich ausgeführt.
- a) Modellieren Sie die n Philosophen in einem rLSTS Γ . Geben Sie für jede Aktion von Γ geeignete Ausführbarkeitsbedingungen an, und beschreiben Sie die Wirkung der Aktionen durch Formeln von \mathcal{L}_{TLF} .
- b) Zeigen Sie: Es gibt einen Ablauf, in dem kein Philosoph jemals isst.

Abgabe: Nach den Ferien: Mittwoch, den 10.1.2007, vor der Übung.