

# **Theoretische Informatik 1**

12. Juni 2013

Praktikumsaufgabe 4

**Tommy Redel**  
**Lucas Jenß**

# 1 Kantine Aufgabe

## 1.1 In der Warteschlange vor der Kasse gibt es maximal 3 Kunden

Für diesen Teil ist lediglich zu prüfen, ob sich auf der Stelle “Warteschlange\_Kasse” auf jedem möglichen Pfad maximal 3 Token befinden:

$$AG \ AF \ Warteschlange\_Kasse \leq 3;$$

## 1.2 Alle 5 Kunden stehen immer wieder zusammen vor der Kantine

Hierfür ist zu prüfen, ob auf allen möglichen Pfaden irgendwann ein Zustand erreicht wird, in dem auf der Stelle “EinGang” 5 Token liegen:

$$AG \ AF \ EinGang == 5;$$

## 1.3 Das Netz ist lebendig

Hier ist zu prüfen, ob jede Transition beliebig oft schalten kann. Da man mit Hilfe von Charlie keine Transitionen direkt überprüfen kann, müssen stattdessen deren Vorbereiche überprüft werden. Dafür wird für jede Transition geprüft, ob in dessen Vorbereich genug Token liegen, um die jeweilige Transition zu schalten.

```
AG (EF(EinGang > 0 & WarteschlangeKapazitaet2 > 0) &  
    EF(EinGang > 0 & WarteschlangeKapazitaet1 > 0) &  
    EF(WarteSchlange2 > 0 & Bedienstete > 0 & AusgabeKapazitaet2) &  
    EF(WarteSchlange1 > 0 & Bedienstete > 0 & AusgabeKapazitaet1) &  
    EF(Ausgabe2 > 0 & KapazitaetKasse > 0) &  
    EF(Ausgabe1 > 0 & KapazitaetKasse > 0) &  
    EF(Warteschlange_Kasse > 0) &  
    EF(SpeiseSaal > 0) &  
    EF(NeueKunden == 5)  
);
```

#### 1.4 Das Netz ist reversibel (Eine selbst formulierte, nicht-triviale Eigenschaft)

Um zu prüfen, ob ein Netz reversibel ist, muss es möglich sein, einen Pfad zu finden auf dem eine Markierung liegt, die der Startmarkierung entspricht.

```
AG EF(  
    EinGang == 5 &  
    WarteschlangeKapazitaet1 == 2 &  
    WarteschlangeKapazitaet2 == 2 &  
    WarteSchlange1 == 0 &  
    WarteSchlange2 == 0 &  
    AusgabeKapazitaet1 == 1 &  
    AusgabeKapazitaet2 == 1 &  
    Ausgabe1 == 0 &  
    Ausgabe2 == 0 &  
    Bedienstete == 1 &  
    KapazitaetKasse == 3 &  
    Warteschlange_Kasse == 0 &  
    SpeiseSaal == 0 &  
    NeueKunden == 0  
    );
```

## 2 Eisenbahn Aufgabe

### 2.1 Ein Zug kann die Acht fahren

Eine Route die eine Acht im Lösungsnetz beschreibt, ist zum Beispiel folgende (die Zahlen stehen hier für die “GA” stellen mit der entsprechenden Nummer):

$$6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 9 \rightarrow 6$$

Dass diese Strecke von einem Zug befahren werden kann, lässt sich zum Beispiel durch folgenden CTL-Ausdruck evaluieren:

$$\begin{aligned} & (AG (GA6 \ \& \ GAK7 \ \& \ SchrankenOffen) \rightarrow EX \ GA7) \ \& \\ & \quad (AG (GA7 \ \& \ GAK8) \rightarrow EX \ GA8) \ \& \\ & \quad (AG (GA8 \ \& \ GAK9 \ \& \ WeicheUnten1) \rightarrow EX \ GA9) \ \& \\ & \quad (AG (GA9 \ \& \ GAK4 \ \& \ WeicheUnten4) \rightarrow EX \ GA4) \ \& \\ & \quad \quad (AG (GA4 \ \& \ GAK3) \rightarrow EX \ GA3) \ \& \\ & \quad \quad (AG (GA3 \ \& \ GAK2) \rightarrow EX \ GA2) \ \& \\ & \quad (AG (GA2 \ \& \ GAK9 \ \& \ WeicheUnten2) \rightarrow EX \ GA9) \ \& \\ & \quad (AG (GA9 \ \& \ GAK6 \ \& \ WeicheUnten3) \rightarrow EX \ GA6); \end{aligned}$$

Es wird mit “GA6” begonnen, also mit jedem Zustand in dem gilt, dass sich ein Token auf “GA6” befindet, und ausserdem die Voraussetzungen erfüllt sind damit sich das Token zu GA7 bewegen kann, also “GAK7 & SchrankenOffen”. Für diese Zustände wird geprüft, ob es einen Folgezustand gibt, sodass “GA7” gilt, d.h. “EX GA7”. Dieses vorgehen wird für alle Stellen auf der oben beschriebenen Route wiederholt, sodass sichergegangen werden kann, dass ein Zug diese Route fahren kann.

## 2.2 Es gilt immer, dass genau dann wenn die Schranke unten ist, kann ein Zug durchfahren

$$AG((GA6 \ \& \ GAK7 \ \& \ SchrankenOffen) \rightarrow EX(GA7)) \ \& \quad (1)$$

$$AG((GA7 \ \& \ GAK6 \ \& \ SchrankenOffen) \rightarrow EX(GA6)); \quad (2)$$

In dieser CTL-Formel werden beide Wege durch die Schranke modelliert, d.h. die Richtung von “GA6” nach “GA7” (1) sowie die Richtung von “GA7” nach “GA6” (2). Es wird jeweils geprüft ob die Voraussetzungen erfüllt sind, dass bspw. der Zug auf “GA6” nach “GA7” fahren kann, und außerdem ob die Schranke offen ist. In jenen Zuständen muss es dann einen Folgezustand geben, für den der Zug durch die offene Schranke gefahren ist.

Man beachte hier, dass die Modellierung unsinnig ist, da wir Schranken für Züge modelliert haben (man stelle sich eine Schrank vor, die die Gleise blockiert). In der Realität ist die Schranke natürlich für die Autos gedacht, sodass der Zug eigentlich nur dann fahren darf, wenn die Schranken geschlossen sind.

## 2.3 Die Schranken sind nie dauerhaft geschlossen

$$AG \ EX \ SchrankenOffen == 1;$$

## 2.4 Das Netz ist 1-sicher

Die folgende CTL-Formel prüft 1-sicherheit für alle Stellen deren Token einen Gleisabschnitt repräsentieren. Wollte man 1-sicherheit für das gesamte Netz prüfen, so müsste man die Formel um die fehlenden Stellen erweitern.

$$\begin{aligned}
&AG (GA1 \leq 1) \ \& \\
&AG (GA2 \leq 1) \ \& \\
&AG (GA3 \leq 1) \ \& \\
&AG (GA4 \leq 1) \ \& \\
&AG (GA5 \leq 1) \ \& \\
&AG (GA6 \leq 1) \ \& \\
&AG (GA7 \leq 1) \ \& \\
&AG (GA8 \leq 1) \ \& \\
&AG (GA9 \leq 1);
\end{aligned}$$

## 2.5 Eine selbst formulierte, nicht-triviale Eigenschaft

Hier wird eine CTL-Formel aufgestellt, welche folgendes sicherstellt:

Wenn alle Schranken oben sind, ist es nicht möglich zu GA9 (mittlerer Gleisabschnitt) zu gelangen.

Zu diesem Zweck werden alle Zustände gesucht, in denen es potenziell möglich wäre auf GA9 zu fahren ("GAK9 & GA9==0"), aber alle Weichen so gestellt sind, dass sie nicht zu GA9 hinführen. Für diese Zustände muss dann gelten, dass auf keinem der Folgezustände GA9 belegt ist, also "AX GA9==0".

$$\begin{aligned}
&AG((GAK9 \\
&\quad \& GA9 == 0 \\
&\quad \& WeicheOben1 \\
&\quad \& WeicheOben2 \\
&\quad \& WeicheOben3 \\
&\quad \& WeicheOben4) \rightarrow AX GA9 == 0);
\end{aligned}$$