

## 10 Endliche Automaten

### 10.1 Erstes Beispiel: ein Getränkeautomat

- siehe Skript;
- Am Freitag in der Übung wird als weiteres Beispiel die Benutzung von Mealy-Automaten für einfache Codierungs- bzw. Decodierungsaufgaben vorkommen.

### 10.2 Mealy-Automaten

- Man nehme den Getränkeautomaten und
  - „überlege“ sich  $f^*((0, -), \mathbf{R10})$  (durch den Zustandsgraphen laufen)
  - „berechne“  $f^*((0, -), \mathbf{R10})$
  - analog  $f^{**}$
- Man erarbeite die alternative Definition

$$f^{**}(z, \varepsilon) = z$$

$$\text{und für alle } x \in X \text{ und } w \in X^* \text{ ist } f^{**}(z, xw) = z \cdot f^{**}(f(z, x), w)$$

- Apropos alternative Definition: In der letzten Klausur galt es per Induktion zu zeigen, dass  $f^*(z, wx) = \bar{f}^*(z, xw)$ . Die Antworten haben mich damals doch etwas traurig gestimmt. Wer nochmal eine Induktion im Tut üben möchte, kann das gerne rechnen lassen.

Man betrachte die folgenden Beispielautomaten:

- Getränkautomat: man mache sich klar:
  - $g^*((0, -), \mathbf{R10}) = \mathbf{R}$
  - $g^{**}((0, -), \mathbf{R10}) = \mathbf{R}$
  - $g^{**}((0, -), \mathbf{R110}) = \mathbf{1R}$
- nur ein Zustand  $z$ ,  $X = Y = \{\mathbf{a}, \mathbf{b}\}$  und  $g(z, \mathbf{a}) = \mathbf{b}$  und  $g(z, \mathbf{b}) = \mathbf{ba}$ 
  - wie sieht  $w_1 = g^{**}(z, \mathbf{a})$  aus?
  - $w_2 = g^{**}(z, w_1), \dots w_{i+1} = g^{**}(z, w_i)$ ?

- was passiert mit den Längen?
- $Z = \mathbb{Z}_5$ ,  $X = \{a, b\}$ ,  $Y = \{0, 1\}$ , bei  $b$  gleicher Zustand, Ausgabe  $0$ , bei  $a$  einen Zustand weiter, bei jedem  $5$ .  $a$  Ausgabe  $1$ , sonst Ausgabe  $0$ . Was tut der Automat?

### 10.3 Moore-Automaten

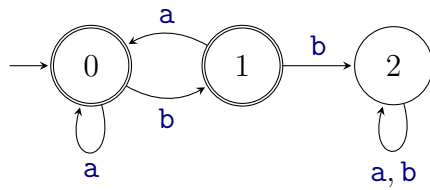
- Die Unterschiede zwischen Moore- und Mealy-Automaten sind „klein“: Abgesehen vom leeren Wort, für das ein Mealy-Automat keine Ausgabe liefern kann, gilt: Man kann zu jedem Moore-Automaten einen Mealy-Automaten konstruieren, so dass das  $g^*$  für beide gleich ist. Und die umgekehrte Richtung von Mealy- zu Moore-Automaten funktioniert auch.
- Falls jemand fragt: Die erste Richtung von Moore zu Mealy ist ganz einfach: Man „zieht die Ausgabe aus einem Zustand „zurück“ zu den Eingaben an den Kanten zu diesem Zustand.

Die umgekehrte Richtung ist ein bisschen aufwändiger, aber auch kein Hexenwerk; siehe <http://de.wikipedia.org/wiki/Mealy-Automat>, Abschnitt *Zusammenhang mit Moore-Automat*.

### 10.4 Endliche Akzeptoren

#### 10.4.1 Beispiele formaler Sprachen, die von endlichen Akzeptoren akzeptiert werden können

- **Bitte bitte bitte die akzeptierenden Zustände nur so nennen**, und *nicht* Endzustände. Langjährige Erfahrung zeigt, dass das zu falschen Intuitionen führt.
- Man entwickle einen Akzeptor mit  $X = \{a, b\}$ , der alle Wörter akzeptiert, bei denen die Anzahl der  $a$  durch  $5$  teilbar ist. (Anzahl der  $b$  ist also egal.)  
Kreis mit  $5$  Zuständen; bei jedem  $a$  eins weiter, bei jedem  $b$  Schlinge; akzeptieren bei Anfangszustand.
- Man entwickle einen Akzeptor mit  $X = \{a, b\}$ , der alle Wörter akzeptiert, in denen nirgends hintereinander zwei  $b$  vorkommen. Hier „muss“ man zählen, wieviele  $b$  unmittelbar hintereinander kamen, aber nur bis  $2$ :



- Diskussion: einfachste Version von Syntaxanalyse

## 10.5 Eine formale Sprache, die von keinem endlichen Akzeptor akzeptiert werden kann

- Der Beweis, dass  $\{a^k b^k \mid k \in \mathbb{N}_0\}$  von keinem endlichen Akzeptor erkannt werden kann, vermittelt ein wesentliches über endliche Automaten: Wenn ein hinreichend langes Wort  $w$  akzeptiert wird (und das ist garantiert immer der Fall, wenn die Sprache unendlich ist), dann läuft man für ein Teilwort  $v$  durch eine Schleife, und dann ändert mehrfaches Durchlaufen der Schleife (bzw. ganz weglassen) nichts am Akzeptierungsverhalten (Pumpinglemma für reguläre Sprachen, das kommt aber erst im dritten Semester).

## 10.6 alte Klausuraufgaben

- Zu Akzeptoren gibt es eigentlich in fast jeder Klausur eine Aufgabe.
- Zu Mealy Automaten habe ich auf die Schnelle Aufgabe 3 gefunden:  
<http://gbi.ira.uka.de/archiv/2009/k-mar10.pdf>