Algorithmen und Datenstrukturen

Theoretische Grundlagen der Informatik

Prof. Dr. rer. nat. Jörg Striegnitz

Fachhochschule Aachen
striegnitz@fh-aachen.de

8. April 2025

Algorithmen und Datenstrukturen

Automaten mit Ausgabe

Prof. Dr. rer. nat. Jörg Striegnitz

Fachhochschule Aachen
striegnitz@fh-aachen.de

8. April 2025

Motivation

Bisher: Automaten zur Lösung von Entscheidungsproblemen (Ausgabe: *true* oder *false*)

Automaten mit Ausgabe:

- **Ausgabeband**
- **Ausgabealphabet** Γ
- Ausgabe erfolgt während Transition (**Mealy-Maschine**) oder bei Eintritt in Zustand (**Moore-Maschine** - hier nicht betrachtet)
- Näher an Berechenbarkeitsbegriff, denn es lassen sich bestimmte Funktionen

  $$\Sigma^* \to \Gamma^*$$

  kodieren

Beispiel: Mealy-Maschine 1

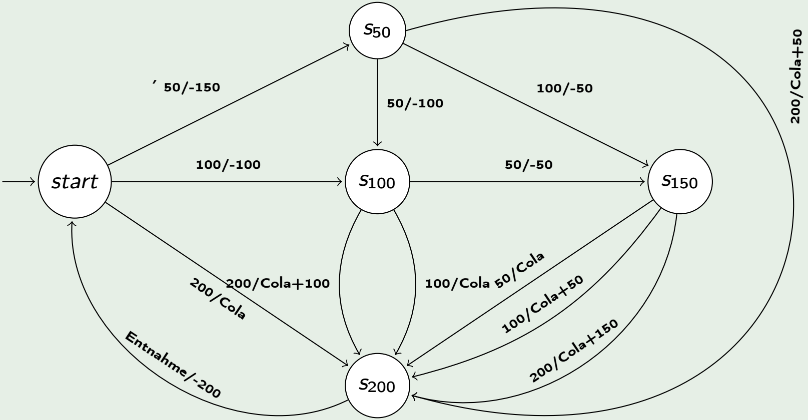
Mealy-Maschine oft zur Beschreibung von Prozessen

Beispiel 1.59 (Getränkeautomat)

- **Hier**: Bedienung Getränkeautomat
- **Benutzeraktionen** als Eingabealphabet Σ:
  - $\underline{50}$, $\underline{100}$, $\underline{200}$: Einwurf des entsprechenden €-Cent-Betrages
  - $\underline{\text{Entnahme}}$: Entnahme Getränk und ggf. des Rückgeldes
- **Maschinen(re)aktionen** als Ausgabealphabet Γ:
  - $\underline{-200}$, $\underline{-150}$, $\underline{-100}$, $\underline{-50}$: Displayanzeige des noch zu zahlenden Betrages
  - $\underline{\text{Cola}}$, $\underline{\text{Cola}+50}$, $\underline{\text{Cola}+100}$, $\underline{\text{Cola}+150}$: Ausgabe von Getränk und ggf. Rückgeld
- **Zustände** als »Speicher« für bereits gezahlten Betrag:
  - $start$, s_{50}, s_{100}, s_{150}, s_{200}
- **Zustandsübergänge**: Paar aus Benutzeraktion/Maschinen(re)aktion

Beispiel: Mealy-Maschine 2

Beispiel 1.60 (Getränkeautomat - Transitionsgraph beschreibt Prozess)



Beachte: Am Anfang keine Ausgabe

Konzeptionell: Anzeige des Automaten zu Beginn leer

Definition 1.65 (Mealy-Maschine)

Eine **Mealy-Maschine** ist ein Sixtupel $M = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$ aus

- einer endlichen **Zustandsmenge** Q,
- einem **Eingabealphabet** Σ,
- einem **Ausgabealphabet** Γ,
- der **Transitionfunktion** $\delta : Q \times \Sigma \to Q$,
- der **Ausgabefunktion** $\lambda : Q \times \Sigma \to \Gamma$ und
- einem **Startzustand** q_0.

- **Beachte**: Mealy-Maschinen haben keine akzeptierenden Zustände!

Mealy-Maschine: Konfiguration / Konfigurationsübergang

In der Konfiguration einer Mealy-Maschine werden deren Zustand, die noch abzuarbeitenden Zeichen (Aktionen) und die bereits erzeugte Ausgabe festgehalten.

Definition 1.66 (Konfigurationsübergang)

Sei $M = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$ eine Mealy-Maschine. Eine **Konfiguration** von M ist ein Element aus $Q \times \Sigma^* \times \Gamma^*$. (q_0, w, ε) heißt **Startkonfiguration**; (q, ε, v) heißt **Endkonfiguration** (für $q \in Q$, $w \in \Sigma^*$ und $v \in \Gamma^*$). Ein ***Konfigurationsübergang*** (ein Schritt) von M ist eine Relation $\underset{M}{\vdash}: (Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Sigma^* \times \Gamma^*)$ mit

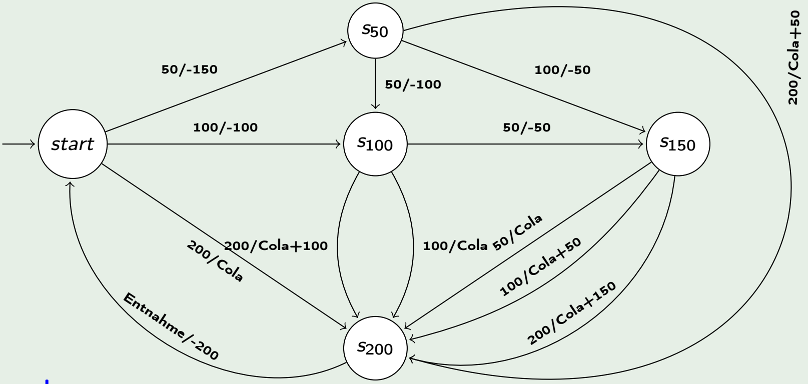
$$(p, av, w) \underset{M}{\vdash} (q, v, wb) \;\Leftrightarrow\; \delta(p, a) = q \text{ und } \lambda(p, a) = b$$

$\underset{M}{\vdash}$ heißt ***Schrittrelation***.

- **Beachte**: In jedem Schritt wird die Ausgabe um ein Zeichen verlängert!
- **Berechnung**: wie DEA, NEA und ε-NEA als Folge von Konfigurationen, die in Schrittrelation stehen. Beachte: hier keine akzeptierenden und keine verwerfenden Berechnungen!

Beispiel: Berechnung

Beispiel 1.61 (Berechnung einer Mealy-Maschine)



Berechnung:

$$(\text{start}, \underline{50}\ \underline{200}\ \underline{\text{Entnahme}}, \varepsilon) \vdash (s_{50}, \underline{200}\ \underline{\text{Entnahme}}, \underline{-150}) \vdash$$
$$(s_{200}, \underline{\text{Entnahme}}, \underline{-150}\ \underline{50+\text{Cola}}) \vdash (\text{start}, \varepsilon, \underline{-150}\ \underline{50+\text{Cola}}\ \underline{-200})$$

Effekt ist Abbildung $\Sigma^* \to \Gamma^*$; hier: $\underline{50}\ \underline{200}\ \underline{\text{Entnahme}} \mapsto \underline{-150}\ \underline{50+\text{Cola}}\ \underline{-200}$

Berechnung einer Mealy-Maschine

- Mealy-Maschine berechnet Funktion

$$f_M : \Sigma^* \to \Gamma^*$$

- Wie beschreibt man f_M?

Definition 1.67 ($\hat{\lambda}$ - Fortsetzung der Ausgabefunktion auf Wörter)

Sei $M = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$ eine Mealy-Maschine. Dann ist $\hat{\lambda} : Q \times \Sigma^* \to \Gamma^*$, die **Fortsetzung von λ auf Wörter**, induktiv definiert durch

- $\hat{\lambda}(q, \varepsilon) = \varepsilon$
- $\hat{\lambda}(q, aw) = \lambda(q, a) \cdot \hat{\lambda}(\delta(q, a), w)$ mit $a \in \Sigma$, $w \in \Sigma^*$ und $q \in Q$.

- Damit kann man f_M leicht beschreiben:

Definition 1.68 (Von einer Mealy-Maschine berechnete Funktion)

Sei $M = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$ eine Mealy-Maschine. Dann ist $f_M : \Sigma^* \to \Gamma^*$, die von M berechnete Funktion, definiert durch

$$f_M(x) = \hat{\lambda}(q_0, x)$$

- Betrachte beliebige Funktion $P : \mathcal{D} \to \mathcal{W}$.
- $\Pi = (P, \mathcal{D}, \mathcal{W})$ nennt man **Problem**.
- Lösung von Π durch Mealy-Maschine $M_\Pi = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$:
  1. Kodierung der Elemente aus $\mathcal{D}$ mittels **Kodierfunktion** $\alpha : \mathcal{D} \to \Sigma^*$
  2. Lauf der Mealy-Maschine (Berechnung von $f_{M_\Pi} : \Sigma^* \to \Gamma^*$) liefert Wort aus Γ^*
  3. Dekodierung der Elemente aus Γ^* mittels **Dekodierfunktion** $\beta : \Gamma^* \to \mathcal{W}$
- Folgendes Diagramm verdeutlicht den Ablauf:

$$\begin{tikzcd}
\Pi & \mathcal{D} \arrow[r, "P"] \arrow[d, "\alpha"'] & \mathcal{W} \\
M_\Pi & \Sigma^* \arrow[r, "f_{M_\Pi}"] & \Gamma^* \arrow[u, "\beta"']
\end{tikzcd}$$

Mealy-berechenbare Probleme 2

Definition 1.69 (Mealy-berechenbare Probleme)

Sei $\Pi = (P, \mathcal{D}, \mathcal{W})$ ein Problem. Eine Mealy-Maschine $M_\Pi = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$ löst das Problem Π, falls es eine Eingabekodierung $\alpha : \Sigma^* \to \mathcal{W}$ und eine Ausgabekodierung $\beta : \Gamma^* \to \mathcal{W}$ gibt, so dass

$$\forall x \in \mathcal{D} \ : \ P(x) = \beta\left(f_{M_\Pi}\left(\alpha(x)\right)\right)$$

gilt. Ein Problem heißt **Mealy-berechenbar**, falls es eine Mealy-Maschine gibt, die das Problem löst.

Achtung! Eigentliche Berechnung von P muss durch M_Π erfolgen.

Beispiel 1.62 (Addition von Binärzahlen)

- Betrachte die Addition von Binärzahlen add : $\mathbb{B}^+ \times \mathbb{B}^+ \to \mathbb{B}^+$ mit $\mathbb{B} = \Sigma_{Bool}$

$$\text{add}(x, y) = \text{Bin}(\text{Zahlwert}(x) + \text{Zahlwert}(y))$$

- Zur Erinnerung:
  - Zahlwert : $\mathbb{B}^+ \to \mathbb{N}$ wandelt Binärzahlen in natürliche Zahlen
  - Bin : $\mathbb{N} \to \mathbb{B}^+$ wandelt natürliche Zahlen in Binärzahlen um
- Addition von Hand:
  - schreibe die zu addierenden Zahlen untereinander
  - addiere Stellenweise von rechts nach links / notiere eventuell entstehende Überträge in der Spalte links von der aktuellen
  - Unbesetzte Stellen nehmen den Wert 0 an

		1	0	1	1	0
+			1	1	1	1
Ü	1	1	1	1		
	1	0	0	1	0	1

Beispiel: Addition von Binärzahlen 2

Beispiel 1.63 (Addition von Binärzahlen - Konstruktionsidee)

$$
\begin{array}{ccccccc}
 & & 1 & 0 & 1 & 1 & 0 \\
+ & & & 1 & 1 & 1 & 1 \\
\text{Ü} & 1 & 1 & 1 & 1 & & \\
\hline
 & 1 & 0 & 0 & 1 & 0 & 1
\end{array}
$$

Umsetzung mit Mealy-Maschine:

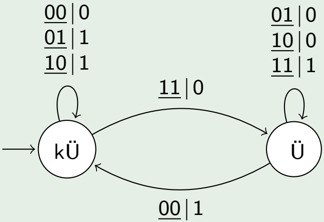
- Mealy-Maschine bekommt Wort aus Zahlenpaaren als Eingabe. Paare ergeben sich aus dem Additionsvorgang - hier also

$$\underline{01}\ \underline{11}\ \underline{11}\ \underline{01}\ \underline{10}\ \underline{00}$$

- Letztes Paar $\underline{00}$ um eventuellen Übertrag am Ende der Berechnung zu berücksichtigen
- Übertrag »merkt« sich die Mealy-Maschine in Zustand - wir brauchen also zwei Zustände.

Beispiel: Addition von Binärzahlen 3

Beispiel 1.64 (Addition von Binärzahlen - Transitionsgraph)



Berechnung:

$$
\begin{aligned}
&(\text{kÜ},\ \underline{01}\ \underline{11}\ \underline{11}\ \underline{01}\ \underline{10}\ \underline{00},\ \varepsilon) \vdash \\
&(\text{kÜ},\ \underline{11}\ \underline{11}\ \underline{01}\ \underline{10}\ \underline{00},\ 1) \vdash \\
&(\text{Ü},\ \underline{11}\ \underline{01}\ \underline{10}\ \underline{00},\ 10) \vdash \\
&(\text{Ü},\ \underline{01}\ \underline{10}\ \underline{00},\ 101) \vdash \\
&(\text{Ü},\ \underline{10}\ \underline{00},\ 1010) \vdash \\
&(\text{Ü},\ \underline{00},\ 10100) \vdash \\
&(\text{kÜ},\ \varepsilon,\ 101001)
\end{aligned}
$$

Offenbar muss das Ergebnis noch gespiegelt werden.

Beispiel: Addition von Binärzahlen 4

Beispiel 1.65 (Addition von Binärzahlen - Kodierung der Eingabe)

Kodierfunktion $\alpha : \mathbb{B}^+ \times \mathbb{B}^+ \to \{\underline{00}, \underline{01}, \underline{10}, \underline{11}\}^+$:

$$\alpha\,(a_{m-1} \cdots a_0\,,\, b_{n-1} \cdots b_0) = \underline{a_0\,b_0}\,\underline{a_1\,b_1} \cdots \underline{a_{m+(n\ominus m)}\,b_{n+(m\ominus n)}}$$

mit

$$k \ominus l = \begin{cases} k - l & \text{falls } k \geq l \\ 0 & \textit{sonst} \end{cases}$$

und $a_m = b_n = 0$ (führende Nullen für eventuellen Übertrag).

Anmerkung: $k \ominus l$ berechnet, um wie viel k größer ist als l, oder liefert 0, falls $k \leq l$

Beachte **Längenausgleich**:

- $a_i = 0$ für $m \leq i \leq m + (n \ominus m)$ und
- $b_i = 0$ für $n \leq i \leq n + (m \ominus n)$.

Beispiel: Addition von Binärzahlen 5

Beispiel 1.66 (Addition von Binärzahlen - Dekodierung der Ausgabe)

- Mealy-Maschine schreibt das Ergebnis spiegelverkehrt auf Ausgabeband.
- Ergebnis enthält eventuell führende Nullen

Dekodierfunktion $\beta : \mathbb{B}^+ \to \mathbb{B}^+$:

$$\beta\left(a_0 \cdots a_k\right) = a_l a_{l-1} \cdots a_0 \text{ falls } z_l = 1 \text{ und } z_i = 0 \text{ für } l + 1 \leq i \leq k$$

- definition: 1.69
- example: 1.66
- solution: 1.6