Theoretische Informatik I

Vorlesung vom 23.06.2000

Skript

Thomas Schmidt
702720

INHALTSVERZEICHNIS

Beispiele:

1) $L = \{ x \in \{0, 1\}^* \mid x \text{ endet mit } 00\}$

$[u] = \{v \in X^* \mid u \equiv_L v\}$

$u \equiv_L v \Leftrightarrow \forall w \in X^*: (uw \in L \Leftrightarrow vw \in L)$

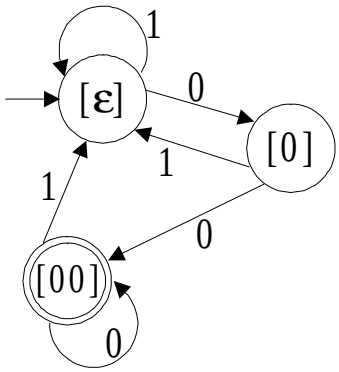
Offenbar gilt:

$[\varepsilon] = \{w \in \{0, 1\}^* \mid w \text{ endet nicht mit } 0\}$

$[0] = \{w \in \{0, 1\}^* \mid w \text{ endet mit 0, aber nicht mit } 00\}$

$[00] = \{w \in \{0, 1\}^* \mid w \text{ endet mit } 00\}$

Dies sind die einzigen Äquivalenzklassen.



2) Sei $L = \{a^n b^n \mid n \geq 0\}$, $X=\{a, b\}$

Die Äquivalenzklassen:

$[\varepsilon] \neq [a]$, denn $\varepsilon ab \in L$, aber $a ab \notin L$

$[a] \neq [aa]$, denn $ab \in L$, aber $aa b \notin L$

$[aa] \neq [aaa]$, denn $aabb \in L$, aber $aaa bb \notin L$

$[a^i] \neq [a^j]$, denn $a^i b^i \in L$, aber $a^j b^i \notin L$

Ergebnis: Man erhält unendlich viele verschiedene Äquivalenzklassen
$\rightarrow$ Der Index von L ist nicht endlich $\rightarrow$ L ist nicht regulär

4.5. Minimierung von Automaten

Myhill-Nerode: Äquivalenzklassenautomat ist der Automat mit der kleinsten Menge von Zuständen (Minimalakzeptor)

Der Beweis besagt:
Für jeden beliebigen Automaten A', der die Sprache L erkennt, gilt:

$$K_{L(A')} \subseteq K_L = K_{A_0}$$

wobei A_0 der Minimalakzeptor ist.

Definition U: Sei A = (X, S, δ, s_0, F) ein endlicher Akzeptor.

Zwei Zustände s, s' $\in$ S heißen äquivalent, wenn für alle w $\in X^*$:

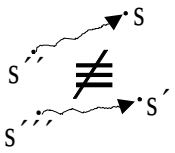
$$\delta(s, w) \in F \Leftrightarrow \delta(s', w) \in F$$

Bemerkung: s und s' können verschmolzen werden und den Automaten verkleinern, ohne dass sich L(A) ändert.

Bemerkung:

a) Offenbar kann man sich bei der Eingabe der Wörter w auf $|w| \leq |s|$ beschränken, da aus Wörtern der Länge $\leq |s'|$ bereits alle Informationen für Definition U gewonnen werden können.
   Hintergrund: Zyklen in Wörtern $>|s|$ und zugehörige Teilwörter können aus w gelöscht werden.

b) Es gilt: $s \in F$, $s' \notin F \Rightarrow$ s ist nicht äquivalent zu s'



Begründung: Setze $w = \varepsilon$ in Definition U

Algorithmus:

Idee: Teste die Äquivalenz von Zuständen ausgehend von den Endzuständen mit inkrementeller Verlängerung der betrachteten Wörter.

Eingabe: deterministischer endlicher Akzeptor A = (X, S, δ, s_0, F), bei dem alle Zustände bereits entfernt sind, die vom Startzustand nicht erreichbar sind.

Ausgabe: Ausgabe, welche Zustände von A zu verschmelzen sind um den minimalen Automaten zu erhalten.

Methode:
1) Bilde Tabelle aller Zustandspaare{s, s'} mit $s \neq s'$
2) Markiere alle Paare {s, s'} mit $s \in F$ und $s' \notin F$ oder umgekehrt
3) Für jedes noch unmarkierte Paar {s, s'} und jedes $a \in X$, test ob $\{\delta(s, a), \delta(s', a)\}$ bereits markiert ist. Wenn ja markiere {s, s'}. (falls $\delta(s, a) = \delta(s', a)$, dann in jedem Fall noch nicht markiert)
4) Wiederhole 3) bis die Tabelle keine Änderung mehr ergibt
5) Alle noch unmarkierten Paare können verschmolzen werden.

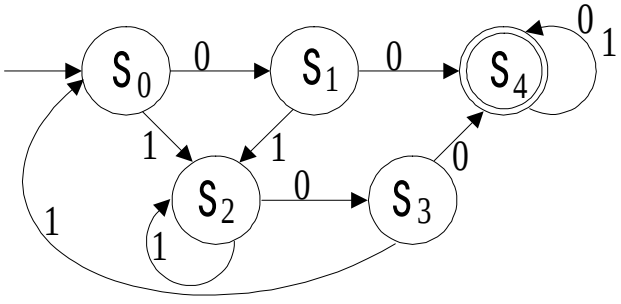
Komplexität: $o(|s|^2)$ bei geeigneter Implementierung

o - Laufzeit

|s| - Anzahl der Zustände

$o(|s|^2) \Rightarrow$ Laufzeit hängt vom Quadrat der Zustände ab

Beispiel:



	s_0	s_1	s_2	s_3
S_1	✳	⊗	⊗	⊗
S_2	①	✳	⊗	⊗
S_3	✳	②	✳	⊗
S_4	×	×	×	×

⊗ Diese werden nicht benötigt, da es keine Verbindungen zwischen diesen Zuständen gibt.
× Diese Zustandspaare sind nach Methode 2) bereits markiert
① $\{s_0\ s_2\}$ ist unmarkiert und kann verschmolzen werden
② $\{s_1\ s_3\}$ ist unmarkiert und kann verschmolzen werden
✳ Diese Zustandspaare sind markiert

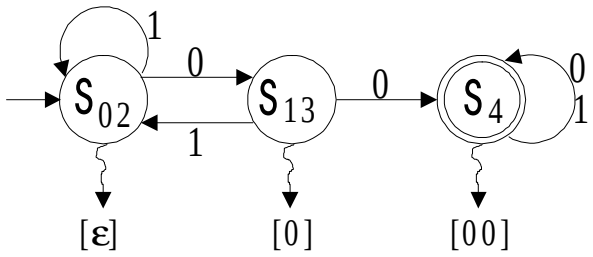
X = {0, 1}

$\{s_0s_3\}$	$\xrightarrow{0}$	$\{s_1s_4\}$	+	$\{s_1s_2\}$	$\xrightarrow{0}$	$\{s_3s_4\}$	+
$\{s_0s_3\}$	$\xrightarrow{1}$	$\{s_0s_2\}$		$\{s_1s_2\}$	$\xrightarrow{1}$	$\{s_2\}$	
$\{s_2s_3\}$	$\xrightarrow{0}$	$\{s_3s_4\}$	+	$\{s_0s_1\}$	$\xrightarrow{0}$	$\{s_1s_4\}$	+
$\{s_2s_3\}$	$\xrightarrow{1}$	$\{s_0s_2\}$		$\{s_0s_1\}$	$\xrightarrow{1}$	$\{s_2\}$	
$\{s_1s_3\}$	$\xrightarrow{0}$	$\{s_4\}$	⊕	$\{s_0s_2\}$	$\xrightarrow{0}$	$\{s_1s_3\}$	⊕
$\{s_1s_3\}$	$\xrightarrow{1}$	$\{s_0s_2\}$	⊕	$\{s_0s_2\}$	$\xrightarrow{1}$	$\{s_2\}$	⊕

\+ da $\{\delta(s, a), \delta(s',a)\}$ für $a \in X$ bereits markiert ist, kann das Paar {s, s'} markiert werden.
⊕ diese Zustandspaare müssen erneut überprüft werden.

$\{s_0s_2\}$	$\xrightarrow{0}$	$\{s_1s_3\}$	
$\{s_0s_2\}$	$\xrightarrow{1}$	$\{s_2\}$	
$\{s_1s_3\}$	$\xrightarrow{0}$	$\{s_4\}$	
$\{s_1s_3\}$	$\xrightarrow{1}$	$\{s_0s_2\}$	

Es ergeben sich keine Änderungen, d.h. $\{s_0s_2\}$ und $\{s_1s_3\}$ können verschmolzen werden.

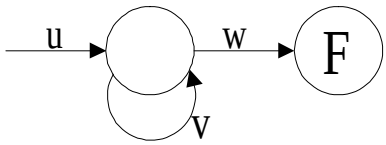


Satz V: (ohne Beweis)
Zu jedem deterministischem endlichem Akzeptor A gibt es einen, bis auf Isomorphie, eindeutig bestimmten minimalen deterministischen endlichen Akzeptor A' mit L(A) = L(A').
„Isomorphie": hier „Gleichgestaltigkeit", Identität bis auf Umbenennung der Zustände.

4.6. Pumping – Lemma für reguläre Sprachen

Zentrales Hilfsmittel zur Entscheidung

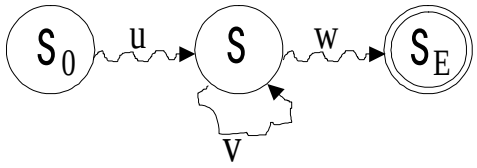
Satz W: Sei X ein Alphabet. Zu jeder regulären Sprache $R \subseteq X^*$ gibt es ein $n \in \mathbb{N}$, so dass für alle Wörter $z \in R$ mit $|z| \geq n$ gilt:
Es gibt eine Zerlegung z = uvw mit $u, v, w \in X^*$.
$v \neq \varepsilon$, $|uv| \leq n$, so dass für alle $i \geq 0$ gilt: $uvw \in R$



$$\forall R \in \mathrm{Reg}\, \exists n \in \mathbb{N} \mid z \mid \geq n \exists u, v, w \in X^* : z = uvw \cap v \neq \varepsilon \wedge \mid uv \mid \leq n : \forall i \in \mathbb{N}_0 : \mathrm{uv}^i \mathrm{w} \in R$$

Idee: Es gibt eine Punktstelle in den Wörtern $z \in R$ ab einer gewissen Länge n.
Warum?

1) endlich Akzeptoren: Um unbeschränkte Wörtern zu erkennen, muss ein Akzeptor in Zyklen laufen. (Zyklen kann man mehrfach durchlaufen)



2) reguläre Ausdrücke: Der einzige Konstruktor um unendliche Sprachen zu erzeugen ist die Sternbildung. Folglich enthalten sehr lange Wörter viele Wiederholungen ein- und desselben Teilwortes.
Beispiel: $\{a^n b^n \mid n \in \mathbb{N}\}$ a^n b^n
u v w
Behauptung: L ist nicht regulär
Beweis: Annahme L ist regulär. Dann gibt es ein $n \in \mathbb{N}$, so dass
$$\forall z \in L |z| \geq n \exists u, v, w \in X^* : z = uvw \cap v \neq \varepsilon \wedge \mid uv \mid \leq n : \forall i \in \mathbb{N}_0 : \mathrm{uv}^i \mathrm{w} \in L$$
Fall1: $u \in \{a\}^*, v \in \{a\}^*, w \in \{b\}^* \Rightarrow$ beim Pumpen ist $uv^i w \notin L$
Fall2: $u \in \{a\}^*, v \in \{b\}^*, w \in \{b\}^* \Rightarrow$ beim Pumpen ist $uv^i w \notin L$
Fall3: $u \in \{a\}^*, v \in \{a\}^*\{b\}^*, w \in \{b\}^* \Rightarrow$ beim Pumpen ist $uv^i w \notin L$