Automaten und Formale Sprachen

SoSe 2007 in Trier

Henning Fernau
Universität Trier
fernau@uni-trier.de

Automaten und Formale Sprachen — Gesamtübersicht

- Organisatorisches
- Einführung
- **Endliche Automaten und reguläre Sprachen**
- Kontextfreie Grammatiken und kontrextfreie Sprachen
- Chomsky-Hierarchie

Endliche Automaten und reguläre Sprachen

1. Deterministische endliche Automaten
2. Nichtdeterministische endliche Automaten
3. Reguläre Ausdrücke
4. **Nichtreguläre Sprachen**
5. Algorithmen mit / für endliche Automaten

Das Pumping-Lemma

Satz: Zu jeder regulären Sprache L gibt es eine Zahl $n > 0$, sodass jedes Wort $w \in L$ mit $\ell(w) \geq n$ als Konkatenation $w = xyz$ dargestellt werden kann mit geeigneten x, y, z mit folgenden Eigenschaften:

1. $\ell(y) > 0$;
2. $\ell(xy) \leq n$;
3. $\forall i \geq 0 : xy^i z \in L$.

Hinweis: Die Umkehrung gilt nicht !

Zur Anwendung des Pumping-Lemmas (schematisch)

1. Wir vermuten, eine vorgegebene Sprache L ist nicht regulär.

2. Im Widerspruch zu unserer Annahme nehmen wir an, L wäre doch regulär. Dann gibt es die im Pumping-Lemma genannte Pumping-Konstante n.

3. Wir wählen ein geeignetes, hinreichend langes Wort $w \in L$ (d.h., $\ell(w) \geq n$). Dies ist der Schritt, wo man leicht "gut" oder "schlecht" wählt ! Bemerkung: Da wir ja vermuten, L ist nicht regulär, ist L insbesondere unendlich, d.h., zu jedem n finden wir ein $w \in L$ mit $\ell(w) \geq n$.

4. Wir diskutieren alle möglichen Zerlegungen $w = xyz$ mit $\ell(y) > 0$ und $\ell(xy) \leq n$ und zeigen für jede solche Zerlegung, dass es ein $i \geq 0$ gibt, sodass $xy^iz \notin L$ gilt.

   Die Komplexität dieser Diskussion hängt im Wesentlichen von der "geschickten" Wahl von w ab; das Anfangsstück von w der Länge n sollte "schön" sein.

Zur Anwendung des Pumping-Lemmas (ein geschickter Einsatz)

Betrachte $L = \{a^k b^k \mid k \in \mathbb{N}\}$.

Wäre L regulär, so gäbe es Pumping-Konstante n.

Betrachte $a^n b^n \in L$; denn: $\ell(a^n b^n) = 2n \geq n$ und Präfix der Länge n ist a^n (sehr schön).

Diskutiere $a^n b^n = xyz$ mit $\ell(xy) \leq n$ und $\ell(y) > 0$:
Offenbar ist $xy \in \{a\}^+$ und damit $y = a^m$ für ein $m > 0$.
$\rightsquigarrow$ Nullpumpen liefert $a^{n-m} b^n \notin L$, $\lightning$ zur Annahme, L wäre regulär.

Zur Anwendung des Pumping-Lemmas (ein ungeschickter Einsatz)

Betrachte $L = \{a^k b^k \mid k \in \mathbb{N}\}$.

Wäre L regulär, so gäbe es Pumping-Konstante n. Da L nur Wörter gerader Länge enthält, können wir annehmen, n ist gerade.

Betrachte $w = a^{n/2} b^{n/2} \in L$ mit $\ell(w) = n$.

Diskutiere $w = xyz$ mit $\ell(xy) \leq n$ und $\ell(y) > 0$:
Fall 1: $xy \in \{a\}^+$ (Nullpumpen ähnlich wie letzte Folie)
Fall 2: $xy = a^{n/2} b^m$ mit $m > 0$.
Fall 2a: $y \in \{b\}^+$ (Nullpumpen ähnlich wie letzte Folie)
Fall 2b: $y = a^r b^m$ mit $r, m > 0$. Dann liegt auch $xy^2z = a^{n/2} b^m a^r b^{n/2} \in L$ ↯ zur Struktur von L.

Die Spiegeloperation

Informell: w^R ergibt sich aus w durch "Rückwärtslesen" (Spiegeln).

Rekursiv: $\lambda^R = \lambda$;
für $w = va$ mit $v \in \Sigma^*$, $a \in \Sigma$ definiere: $w^R := a(v^R)$.

Beispiel: $(abcd)^R = d(abc)^R = dc(ab)^R = dcb(a)^R = dcba(\lambda^R) = dcba\lambda = dcba$.

Erweiterung auf Wortmengen: $L^R = \{w^R \mid w \in L\}$.

Satz: Die regulären Sprachen sind unter Spiegelung abgeschlossen.

Beweis: Wichtig: Wahl des richtigen Modells!

Noch eine Anwendung des Pumping-Lemmas

Betrachte $L = \{w \in \{a, b\}^* \mid w = w^R\}$ (*Palindrome*)

Wäre L regulär, so gäbe es Pumping-Konstante n für L.

Betrachte $w = a^n b a^n \in L$ mit $\ell(w) \geq n$.

Widerspruch ergibt sich durch Nullpumpen.

. . . und noch eine . . .

Betrachte $L = \{a^{k^2} \mid k \in \mathbb{N}\}$.

Wäre L regulär, so gäbe es Pumping-Konstante n für L.

Betrachte $w = a^{(n+1)^2} \in L$ mit $\ell(w) \geq n$.

Widerspruch ergibt sich durch Nullpumpen:
Genauer haben wir, dass für ein $0 < i \leq n$ stimmen muss, dass $a^{(n+1)^2-i} \in L$ gilt, im Widerspruch zu folgender Abschätzung, die $a^{(n+1)^2-i} = a^{r^2}$ annimmt:

$$r^2 \leq n^2 < n^2 + n + (n - i) + 1 = n^2 + 2n + 1 - i = (n + 1)^2 - i.$$

Der Beweis des Pumping-Lemmas

Ist L endlich, so ist die Aussage trivial mit $n := \max\{\ell(w) \mid w \in L\}$.

Ist L unendlich aber regulär, so wird L von einem DEA A mit n Zuständen akzeptiert mit Anfangszustand q_0.

Betrachte ein Wort $w = a_1 \dots a_m \in L$, $a_i \in \Sigma$, $m \geq n$.
Sei $(q_k, a_{k+1} \dots a_m)$ für $0 \leq k \leq n$ die Konfiguration nach k Schritten von A.
Da hierbei $n + 1$ Zustände durchlaufen werden, gibt es nach dem Schubfachprinzip einen Zustand, der zweimal erreicht wird, d.h., $\exists 0 \leq r < s \leq n : q_r = q_s$.
$\rightsquigarrow y = a_{r+1} \dots a_s$ erfüllt $(q_r, y) \vdash_A^* (q_r, \lambda)$.
$\rightsquigarrow \forall i \geq 0 : (q_r, y^i) \vdash_A^* (q_r, \lambda)$.
Mit $x = a_1 \dots a_r$ und $z = a_{s+1} \dots a_m$ folgt die Behauptung.

Nicht-Regularität durch Abschlusseigenschaften

Beispiel: Betrachte die Menge $L \subseteq \{a, b\}^*$ mit der Eigenschaft, dass $w \in L$ liegt gdw. w gleich viele a's wie b's besitzt.

Behauptung: L ist nicht regulär.

Beweis durch Widerspruch: Wäre L regulär, so auch $L' = L \cap \{a\}^*\{b\}^*$, denn $\{a\}^*\{b\}^*$ ist regulär, und der Schnitt zweier regulärer Sprachen ist wiederum regulär.
Offenbar gilt: $L' = \{a^k b^k \mid k \in \mathbb{N}\}$, und von dieser Sprache wissen wir bereits, dass sie nicht-regulär ist.
↯ zu unserer Annahme, L wäre regulär.

Auch hier Schwierigkeit: "Geschickte" Wahl der Operation...

Äquivalenzrelationen (hoffentlich noch bekannt ?!)

Eine Relation $R \subseteq X \times X$ heißt *Äquivalenzrelation* gdw.
(1) $R^0 = \Delta_X \subseteq R$ (Reflexivität)
(2) $R^2 = R \circ R \subseteq R$ (Transitivität)
(3) Mit $R^{-1} = \{(y, x) \mid (x, y) \in R\}$ gilt $R^{-1} \subseteq R$ (Symmetrie)

Eine ÄR auf X induziert eine *Partition* von X
in *Äquivalenzklassen* $[x]_R = \{y \in X \mid xRy\}$.

Eine Äquivalenzrelation auf Σ^*

Es sei $h : (\Sigma^*, \cdot, \lambda) \to (M, \circ, e)$ ein Monoidmorphismus.
Dann ist $x \equiv_h y$ gdw. $h(x) = h(y)$ eine Äquivalenzrelation auf Σ^*.
(Es ist klar, dass diese Relation reflexiv, symmetrisch und transitiv ist ?!)

Noch eine Äquivalenzrelation auf Σ^*

Es sei $A = (Q, \Sigma, \delta, q_0, F)$ ein DEA.
Definiere $u \equiv_A v$ gdw. $\exists q \in Q : ((q_0, u) \vdash_A^* (q, \lambda)) \wedge ((q_0, v) \vdash_A^* (q, \lambda))$.
(Es ist klar, dass diese Relation reflexiv, symmetrisch und transitiv ist ?!)

. . . und noch eine Äquivalenzrelation auf Σ^*

Es sei $L \subseteq \Sigma^*$. L *trennt* zwei Wörter $u, v \in \Sigma^*$ gdw. $|\{u, v\} \cap L| = 1$.

Zwei Wörter u und v heißen *kongruent modulo* L (i.Z.: $u \equiv_L v$), wenn für jedes beliebige Wort w aus Σ^* die Sprache L die Wörter uw und vw *nicht* trennt, d.h. wenn gilt:

$$(\forall w \in \Sigma^*)\ (\ uw \in L \Leftrightarrow vw \in L\ )$$

Satz: Für jede Sprache $L \subseteq \Sigma^*$ ist $\equiv_L$ eine Äquivalenzrelation.

Beweis: Reflexivität: $\forall u \in \Sigma^* : u \equiv_L u$ ✓

Symmetrie: $\forall u, v \in \Sigma^* : u \equiv_L v \Rightarrow v \equiv_L u$ ✓

Transitivität: $\forall u, v, x \in \Sigma^* : (u \equiv_L v \wedge v \equiv_L x) \Rightarrow u \equiv_L x$

Betrachte $u, v, x, w \in \Sigma^*$.

(a) Falls $uw \in L$, so auch $vw \in L$, denn $u \equiv_L v$; wegen $v \equiv_L x$ gilt daher $xw \in L$, d.h., $u \equiv_L x$.

(b) Falls $uw \notin L$, ... (analog)

Beispiel: Betrachte

$$L = \{a^k b^k | k > 0\}$$

$a^i b \not\equiv_L a^j b$ für $i \neq j$:
Verwende $w = b^{i-1}$ mit $a^i b w \in L$ und $a^j b w \notin L$.
Damit hat man für $i = 1, 2, 3, ...$ bereits unendlich viele verschiedene Äquivalenzklassen $[a^i b]$ gefunden.
Genauer gilt: $[a^i b] = \{a^i b, a^{i+1} b^2, a^{i+2} b^3, ...\}$.
Ferner gilt: $[ab] = L$.

Lemma: Es sei $L \subseteq \Sigma^*$ regulär, d.h., L ist durch ein endliches Monoid $(M, \circ, e)$, einen Monoidmorphismus $h : \Sigma^* \to M$ und eine endliche Menge $F \subseteq M$ beschrieben. Dann gilt: Falls $u \equiv_h v$, so $u \equiv_L v$.

Beweis: Betrachte zwei Wörter $u, v \in \Sigma^*$ mit $u \equiv_h v$, also $h(u) = h(v)$.

Da h Morphismus, ist für $w \in \Sigma^*$: $h(uw) = h(u) \circ h(w) = h(v) \circ h(w) = h(vw)$.

Also liegen entweder sowohl uw als auch vw in L oder beide nicht.

Daher gilt $u \equiv_L v$.

Hinweis: Ähnlicher Beweis über DEA-Äquivalenz $\equiv_A$!

Folgerung: Ist L regulär, so hat $\equiv_L$ nur endlich viele Äquivalenzklassen.

Noch mehr Folgerungen aus dem letzten Beweis:

Betrachte reguläre Sprache $L \subseteq \Sigma^*$ und sie beschreibene Homomorphismen h bzw. Automaten A:

Ist $\mathcal{L} := \{L_1, \ldots, L_n\}$ die durch $\equiv_L$ induzierte Partition von Σ^*, so gilt für die durch $\equiv_h$ induzierte Partition $\mathcal{H} := \{H_1, \ldots, H_m\}$ von Σ^* (bzw. für die durch $\equiv_A$ induzierte Partition $\mathcal{A} := \{A_1, \ldots, A_\ell\}$ von Σ^*):
Für jedes H_i (bzw. A_i) gibt es ein L_j mit $H_i \subseteq L_j$ (bzw. $A_i \subseteq L_j$).
Daher heißen $\mathcal{H}$ und $\mathcal{A}$ auch *Verfeinerung*en von $\mathcal{L}$.

Satz: [Myhill und Nerode] Eine Sprache $L \subseteq \Sigma^*$ ist genau dann regulär, wenn es nur endlich viele Äquivalenzklassen bezüglich $\equiv_L$ gibt.

Beweis: 1. L regulär $\Rightarrow$ L hat endlich viele Äquivalenzklassen (siehe Folgerung).

2. Umkehrung: Sei k Zahl der Klassen von $\equiv_L$, d.h. $\Sigma^* = [x_1] \cup ... \cup [x_k]$.
Definiere den *Minimalautomaten* $A(L) = (S, \Sigma, \delta, s_0, F)$ durch
$Q = \{[x_1], ..., [x_k]\}$
$q_0 := [\lambda]$
F bestehe aus allen Äquivalenzklassen $[x_i]$ mit $x_i \in L$
$\delta([x], a) := [xa]$

Wichtig: Mit $[x] = [y]$ ist $xaw \in L \Leftrightarrow yaw \in L$,
also auch $[xa] = [ya]$, $\rightsquigarrow$

$$\delta([x], a) = [xa] = [ya] = \delta([y], a)$$

$\rightsquigarrow$ δ ist wohldefiniert!

Offensichtlich gilt $([\lambda], x) \vdash_A^* ([x], \lambda)$ $\rightsquigarrow$

$$x \in L(A) \Longleftrightarrow \exists q \in F : ([\varepsilon], x) \vdash_A^* (q, \lambda) \Longleftrightarrow [x] \in F \Longleftrightarrow x \in L$$

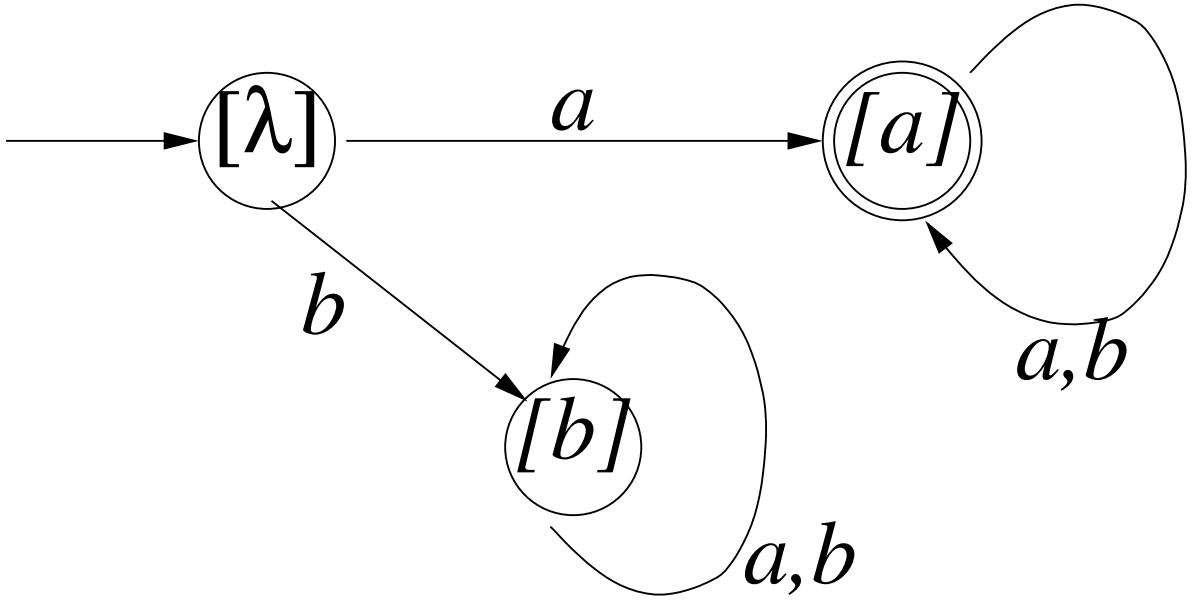
Beispiel: Betrachte $L = \{a\}\{b, a\}^*$

- $\lambda \not\equiv_L a$ mit $\lambda\lambda = \lambda \notin L$, $a\lambda = a \in L$.
- $b \not\equiv_L a$ mit $b\lambda = b \notin L$, $a\lambda = a \in L$
- $\lambda \not\equiv_L b$ mit $\lambda a = a \in L$, $ba \notin L$.
- $bw \equiv_L b$
- $aw \equiv_L a$

Also gilt

$$\Sigma^* = [\lambda] \cup [b] \cup [a]$$

mit dem Minimalautomaten



Warum heißt der Minimalautomat so?

Lemma: Ist L regulär, so ist $A(\mathrm{L})$ der DEA mit der kleinsten Anzahl von Zuständen.

Beweis: Zunächst sieht man: $\equiv_{\mathrm{L}} = \equiv_{A(\mathrm{L})} \rightsquigarrow$
Zustände von $A(\mathrm{L})$ ist gleich # Äquivalenzklassen von $\equiv_{\mathrm{L}}$.

Aus dem Beweis von obigem Lemma lesen wir ab:

Ist A ein DEA mit $\mathrm{L} = \mathrm{L}(A)$, so ist:

Zustände von A ist gleich

Äquivalenzklassen von $\equiv_A$ ist größer gleich

Äquivalenzklassen von $\equiv_{\mathrm{L}}$.

Es gibt nur einen Minimalautomaten

Lemma: Der Minimalautomat ist “bis auf Isomorphie” (Umbenennen der Zustände) eindeutig bestimmt.

Beweis: Es sei L regulär und n die Zustandsanzahl von $A(L)$ sowie die eines evtl. anderen DEA $A = (Q, \Sigma, \delta, q_0, F)$ mit $L(A) = L$.
(Erinnerung: allgemein gilt $|Q(A)| \geq n$ für DEAs A mit $L(A) = L$, denn $\equiv_A$ ist eine Verfeinerung von $\equiv_L$.)
Gilt nun sogar $|Q| = n$, so ist $\equiv_L = \equiv_A$.
$\rightsquigarrow x \equiv_L y \iff x \equiv_A y \iff \exists q \in Q : ((q_0, x) \vdash_A^* (q, \lambda)) \wedge ((q_0, y) \vdash_A^* (q, \lambda))$ für alle $x, y \in \Sigma^*$.
Definiere $\phi : Q \to 2^{\Sigma^*}, q \mapsto \{w \in \Sigma^* \mid (q_0, w) \vdash_A^* (q, \lambda)\}$. ϕ identifiziert die Zustände von A mit den Äquivalenzklassen von $\equiv_A$ und somit mit denen von $\equiv_L$. Anfangs- und Endzustände werden erhalten. Für irgendein Wort $w_q \in \phi(q)$ gilt: (1) $([w_q]_L, a) \vdash_{A(L)} ([w_q a]_L, \lambda)$ sowie (2) $(q, a) \vdash_A (q', \lambda)$ mit $\phi(q') = [w_q a]_L$. Daher wird auch die Übergangsfunktion mit ϕ erhalten, liefert also insgesamt den behaupteten Automatenmorphismus.

Automatenmorphismen

Es seien $A_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ und $A_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$ DEA.
Eine Funktion $f : Q_1 \to Q_2$ heißt *Automatenmomorphismus* von A_1 nach A_2 gdw.:

- Für alle $a \in \Sigma$ und für alle $q \in Q_1$ gilt $f(\delta_1(q, a)) = \delta_2(f(q), a)$.
- $f(q_{01}) = q_{02}$.
- Für alle $q \in Q_1$ gilt: $q \in F_1 \iff f(q) \in F_2$.

Gibt es so einen Morphismus, so gilt $L(A_1) = L(A_2)$.
Ist f bijektiv, ist f ein *Automatenisomorphismus*.

Eine Anwendung des Satzes von Myhill und Nerode

Folgerung: Hat $\equiv_L$ unendlich viele Äquivalenzklassen, so ist L nicht regulär.

Beispiel: Zu

$$L = \{a^k b^k | k > 0\}$$

hat $\equiv_L$ unendlich viele Äquivalenzklassen, ist also nicht regulär.