

---

# Grundlagen der theoretischen Informatik

---

Kurt Sieber

Fakultät IV, Department ETI  
Universität Siegen

SS 2013

Vorlesung vom 30.04.2013

# Reguläre Sprachen

---

## Grenzen regulärer Sprachen

Wie beweist man, dass eine Sprache *nicht* regulär ist?

Oder allgemeiner:

Wie findet man heraus, *ob* eine Sprache regulär ist oder nicht?

Wir hatten schon ein Beispiel—nämlich  $L = \{a^n b^n \mid n \geq 0\}$ —mit dem *Schubfachprinzip* bewiesen.

Jeder DEA  $A$  verteilt die Wörter  $w \in \Sigma^*$  auf endlich viele ‘Schubladen’, nämlich auf seine Zustände.

Wenn wir also *unendlich viele* Wörter finden, von denen je zwei *nicht* in der gleichen Schublade stecken dürfen, so kann kein DEA für die Sprache existieren.

Bei der Sprache  $L$  waren das die Wörter  $a^n$  mit  $n \geq 0$ .

## Reguläre Sprachen

---

Die Annahme, dass zwei dieser Wörter,  $a^i$  und  $a^j$ , in ein- und demselben Zustand landen, führt zum Widerspruch,

denn dann würden auch  $a^i b^i$  und  $a^j b^i$  in einem gemeinsamen Zustand landen, obwohl  $a^i b^i \in L$  und  $a^j b^i \notin L$ .

**Definition 2.28** Sei  $L \subseteq \Sigma^*$ . Zwei Wörter  $u_1, u_2 \in \Sigma^*$  heißen *L-unterscheidbar*, wenn ein Wort  $v \in \Sigma^*$  existiert mit  $u_1 v \in L$  und  $u_2 v \notin L$  oder umgekehrt (d.h. wenn man sie nicht in die gleiche Schublade stecken darf).

Damit erhalten wir das folgende

**Beweisprinzip:** Um zu zeigen, dass eine Sprache  $L$  nicht regulär ist, genügt es, *unendlich viele* Wörter in  $\Sigma^*$  zu finden, die *paarweise L-unterscheidbar* sind.

# Reguläre Sprachen

---

## Beispiele nicht regulärer Sprachen:

### 1. “Ein EA kann nicht zählen”

- $L = \{a^n b^n \mid n \geq 0\}$ : Die Wörter  $a^n$  mit  $n \geq 0$  sind paarweise  $L$ -unterscheidbar, da  $a^i b^i \in L$  und  $a^j b^i \notin L$  für alle  $i \neq j$ .
- $L_1 = \{w \in \{a, b\}^* \mid \#_a(w) = \#_b(w)\}$ : Die Wörter  $a^n$  mit  $n \geq 0$  sind paarweise  $L_1$ -unterscheidbar, da  $a^i b^i \in L_1$  und  $a^j b^i \notin L_1$  für alle  $i \neq j$ .
- $L_2 = \{a^n b^m \mid 0 \leq n < m\}$ : Die Wörter  $a^n$  mit  $n \geq 0$  sind paarweise  $L_2$ -unterscheidbar, denn: Wenn  $i \neq j$ , dann dürfen wir  $i < j$  annehmen (weil  $L$ -Unterscheidbarkeit eine symmetrische Relation ist) und erhalten  $a^i b^{i+1} \in L_2$ , aber  $a^j b^{i+1} \notin L_2$  weil  $j \geq i + 1$ .
- $L_3 = \{a^n b^m \mid 0 \leq n < 2^m\}$ : Die Wörter  $a^{2^n - 1}$  mit  $n \geq 0$  sind paarweise  $L_3$ -unterscheidbar, denn: Für alle  $i < j$  gilt  $a^{2^i - 1} b^i \in L_2$  weil  $2^i - 1 < 2^i$  und  $a^{2^j - 1} b^i \notin L_2$  weil  $2^j - 1 \geq 2^i + 2^i - 1 \geq 2^i$ .

## Reguläre Sprachen

---

### 2. “Ein EA kann sich keine beliebig großen Wörter merken”

- $L_4 = \{ww \mid w \in \{a, b\}^*\}$ : Die Wörter  $a^n b$  mit  $n \geq 0$  sind paarweise  $L_4$ -unterscheidbar, denn: Für alle  $i \neq j$  ist  $a^i b a^i b \in L_4$  und  $a^j b a^i b \notin L_4$ . (Man kann auch die Wörter  $a^n$  betrachten, denn  $a^i$  und  $a^j$  lassen sich durch  $b a^i b$  unterscheiden.)
- $L_5 = \{w \in \{a, b\}^* \mid w = w^R\}$  :

s. Übung

### 3. “Eine reguläre Sprache kann keine beliebig großen Lücken enthalten”

- $L_5 = \{a^{n^2} \mid n \geq 0\}$  :

s. Übung

## Reguläre Sprachen

---

- $L_6 = \{a^{2^n} \mid n \geq 0\}$  : Die Wörter aus  $L_6$  sind paarweise  $L_6$ -unterscheidbar, denn für alle  $i < j$  gilt  $a^{2^i} a^{2^i} = a^{2^i+2^i} = a^{2^{i+1}} \in L_6$  und  $a^{2^j} a^{2^i} = a^{2^j+2^i} \notin L_6$ , weil  $2^j + 2^i < 2^j + 2^j = 2^{j+1}$  zwischen den beiden aufeinanderfolgenden Zweierpotenzen  $2^j$  und  $2^{j+1}$  liegt und deshalb selbst *keine* Zweierpotenz sein kann.
- $L_7 = \{a^p \mid p \text{ ist Primzahl}\}$ :

Wir wollen beweisen, dass zwei *beliebige* Wörter  $a^i, a^j \in \{a\}^*$   $L_7$ -unterscheidbar sind, wobei wir wieder  $i < j$  annehmen dürfen.

Dazu brauchen wir eine Zahl  $k$  mit  $a^j a^k \in L_7$  und  $a^i a^k \notin L_7$ , d.h.  $j + k$  ist Primzahl und  $i + k$  nicht.

Um ein solches  $k$  zu finden, genügt es zu wissen, dass die Menge der Primzahlen beliebig große Lücken enthält, d.h. für jedes  $n > 0$  existieren  $n$  aufeinanderfolgende Zahlen, die keine Primzahlen sind.

## Reguläre Sprachen

---

Sei  $p$  die kleinste Primzahl, die über einer solchen Lücke der Größe  $j$  liegt.

Dann sind (mindestens) die Zahlen  $p-j, \dots, p-1$  *keine* Primzahlen.

Also können wir  $k = p-j$  wählen, denn  $j+k = p$  ist eine Primzahl und  $i+k = i+p-j = p-(j-i)$  fällt in die Lücke und ist deshalb *keine* Primzahl.

Die Existenz beliebig großer Lücken in der Menge der Primzahlen lässt sich leicht einsehen:

Eine Lücke der Größe  $n$  bilden z.B. die Zahlen  $(n+1)!+2, \dots, (n+1)!+n+1$ , denn:

$(n+1)!$  ist durch jede der Zahlen  $2, \dots, n+1$  teilbar, also ist  $(n+1)!+m$  durch  $m$  teilbar für jedes  $m \in \{2, \dots, n+1\}$ , und damit *keine* Primzahl.

## Reguläre Sprachen

---

Eine alternative Formulierung des Beweisprinzips

**Definition 2.29** Sei  $L \subseteq \Sigma^*$ . Zwei Wörter  $u_1, u_2 \in \Sigma^*$  heißen *L-äquivalent* ( $u_1 \sim_L u_2$ ), wenn sie nicht *L-unterscheidbar* sind, d.h. wenn für jedes  $v \in \Sigma^*$  entweder  $u_1v, u_2v \in L$  oder  $u_1v, u_2v \notin L$ .

Wir schreiben  $\not\sim_L$  für die Verneinung von  $\sim_L$ , d.h.  $u_1 \not\sim_L u_2$  bedeutet, dass  $u_1$  und  $u_2$  *L-unterscheidbar* sind.

Für jede Sprache  $L \subseteq \Sigma^*$  ist  $\sim_L$  eine Äquivalenzrelation auf  $\Sigma^*$  (reflexiv, transitiv und symmetrisch). Das sieht man am besten, wenn man die Definition von  $\sim_L$  etwas umformuliert: Für jedes  $u \in \Sigma^*$  sei

$$\text{Erg}_L(u) = \{v \in \Sigma^* \mid uv \in L\}$$

die Menge der *L-Ergänzungen* von  $u$ . Dann gilt

$$\begin{aligned} u_1 \sim_L u_2 &\Leftrightarrow \text{Erg}_L(u_1) = \text{Erg}_L(u_2) \\ &\Leftrightarrow \text{für alle } v \in \Sigma^* \text{ gilt : } v \in \text{Erg}_L(u_1) \Leftrightarrow v \in \text{Erg}_L(u_2) \\ &\Leftrightarrow \text{für alle } v \in \Sigma^* \text{ gilt : } u_1v \in L \Leftrightarrow u_2v \in L \end{aligned}$$



## Reguläre Sprachen

---

### Intuition:

Die  $L$ -Ergänzungen von  $u$  sind genau die Restwörter, die das Anfangswort  $u$  in die Sprache  $L$  überführen, also genau die Wörter, die man 'noch erwartet', wenn man bereits  $u$  gelesen hat.

Also sind zwei Wörter  $u_1, u_2$  genau dann  $L$ -äquivalent, wenn man nach Einlesen von  $u_1$  und  $u_2$  die gleichen Restwörter erwartet, d.h. wenn man  $u_1$  und  $u_2$  in die gleiche 'Schublade' packen darf.

### Schreibweise:

Mit  $[u]_L$  bezeichnen wir die  *$L$ -Äquivalenzklasse* eines Wortes  $u \in \Sigma^*$ , d.h.

$$[u]_L = \{u' \in \Sigma^* \mid u' \sim_L u\}$$

Es gilt also

$$\begin{aligned} u_1 \sim_L u_2 &\Leftrightarrow [u_1]_L = [u_2]_L \\ u_1 \not\sim_L u_2 &\Leftrightarrow [u_1]_L \neq [u_2]_L \end{aligned}$$

## Reguläre Sprachen

---

Damit können wir unser **Beweisprinzip** neu formulieren:

Um zu zeigen, dass eine Sprache  $L$  *nicht* regulär ist, genügt es, unendlich viele Äquivalenzklassen  $[u]_L$  zu finden

(denn unendlich viele paarweise  $L$ -unterscheidbare Wörter sind nichts anderes als die Vertreter von unendlich vielen Äquivalenzklassen).

Wir wollen zeigen, dass auch die Umkehrung gilt, d.h. dass eine Sprache  $L$  regulär ist, wenn es nur endlich viele  $L$ -Äquivalenzklassen gibt.

Beides wird zusammengefasst im **Satz von Myhill und Nerode**, zu dessen Vorbereitung wir zunächst noch einige Eigenschaften der Relation  $\sim_L$  beweisen.

## Reguläre Sprachen

---

### Lemma 2.30

1. Für alle  $u_1, u_2, v \in \Sigma^*$  gilt: Wenn  $u_1 \sim_L u_2$ , dann  $u_1v \sim_L u_2v$ .

(Eine Äquivalenzrelation mit dieser Eigenschaft bezeichnet man als **Rechtskongruenzrelation**.)

2. Für jedes  $u \in \Sigma^*$  gilt:  $u \in L \Leftrightarrow [u]_L \subseteq L$ . (Also  $L = \bigcup_{u \in L} [u]_L$ .)

### Beweis:

1. Seien  $u_1, u_2, v \in \Sigma^*$  und  $u_1 \sim_L u_2$ . Dann gilt für alle  $v' \in \Sigma^*$ :

$$(u_1v)v' \in L \Leftrightarrow u_1(vv') \in L \Leftrightarrow u_2(vv') \in L \Leftrightarrow (u_2v)v' \in L$$

Also ist  $u_1v \sim_L u_2v$ .

2. '⇐' ist klar.

'⇒': Sei  $u \in L$  und  $u' \sim_L u$ .

Wegen  $u\varepsilon \in L$  ist dann auch  $u'\varepsilon \in L$ , also  $u' \in L$ . □

---

## Reguläre Sprachen

---

**Satz 2.31 (Satz von Myhill und Nerode)** *Eine Sprache  $L \subseteq \Sigma^*$  ist genau dann regulär, wenn die Äquivalenzrelation  $\sim_L$  nur endlich viele Äquivalenzklassen besitzt.*

### Beweis:

' $\Rightarrow$ ': Sei  $L \subseteq \Sigma^*$  regulär.

Sei  $A = (\Sigma, Q, s, F, \delta)$  ein DEA mit  $L(A) = L$

und seien  $u_1, u_2 \in \Sigma^*$  mit  $\delta^*(s, u_1) = \delta^*(s, u_2)$ .

Dann gilt für alle  $v \in \Sigma^*$ :

$$\delta^*(s, u_1v) = \delta^*(\delta^*(s, u_1), v) = \delta^*(\delta^*(s, u_2), v) = \delta^*(s, u_2v),$$

$$\text{also } u_1v \in L \Leftrightarrow \delta^*(s, u_1v) \in F \Leftrightarrow \delta^*(s, u_2v) \in F \Leftrightarrow u_2v \in L.$$

Das bedeutet  $u_1 \sim_L u_2$ .

## Reguläre Sprachen

---

Damit haben wir bewiesen, dass zwei Wörter, die in  $A$  zum gleichen Zustand führen, stets  $L$ -äquivalent sind.

Also besitzt  $\sim_L$  nur endlich viele Äquivalenzklassen, nämlich höchstens  $n$ , wobei  $n$  die Anzahl der erreichbaren Zustände von  $A$  ist.

‘ $\Leftarrow$ ’: Sei  $L \subseteq \Sigma^*$  eine Sprache, die nur endlich viele  $L$ -Äquivalenzklassen besitzt.

Da  $L$ -äquivalente Wörter in die gleiche ‘Schublade’ gesteckt werden können, definieren wir einen DEA, der für jede  $L$ -Äquivalenzklasse eine passende Schublade besitzt.

Deshalb nehmen wir die  $L$ -Äquivalenzklassen selbst als Schubladen, d.h. als Zustände eines DEA.

## Reguläre Sprachen

---

Sei  $A = (\Sigma, Q, s, F, \delta)$  mit:

$Q = \{[u]_L \mid u \in \Sigma^*\}$  (die endliche Menge der  $L$ -Äquivalenzklassen)

$s = [\varepsilon]_L$

$F = \{[u]_L \mid u \in L\}$  (=  $\{[u]_L \mid [u]_L \subseteq L\}$  nach Lemma 2.30)

$\delta : Q \times \Sigma \rightarrow Q$      $\delta([u]_L, a) = [ua]_L$

Man beachte, dass  $\delta$  wohldefiniert ist, d.h. dass  $[ua]_L$  unabhängig von der Wahl des speziellen Vertreters  $u \in [u]_L$  ist.

Wenn nämlich  $u' \sim_L u$  ein anderes Element aus  $[u]_L$  ist, so folgt  $u'a \sim_L ua$  und damit  $[u'a]_L = [ua]_L$ .

Um zu zeigen, dass  $A$  tatsächlich die Sprache  $L$  erkennt, beweisen wir, dass jedes Wort  $w \in \Sigma^*$  in die passende Schublade gerät, d.h. dass

$$\delta^*(s, w) = [w]_L \quad \text{für alle } w \in \Sigma^* \quad (*)$$

## Reguläre Sprachen

---

$w = \varepsilon$ :

$$\begin{aligned}\delta^*(s, \varepsilon) &= s && \text{per Definition von } \delta^* \\ &= [\varepsilon]_L && \text{per Definition von } s\end{aligned}$$

$w = va$ :

$$\begin{aligned}\delta^*(s, va) &= \delta(\delta^*(s, v), a) && \text{per Definition von } \delta^* \\ &= \delta([v]_L, a) && \text{nach Induktionsannahme für } v \\ &= [va]_L && \text{per Definition von } \delta\end{aligned}$$

Damit erhalten wir:

$$\begin{aligned}w \in L(A) &\Leftrightarrow \delta^*(s, w) \in F && \text{per Definition von } L(A) \\ &\Leftrightarrow [w]_L \in F && \text{wegen } (*) \\ &\Leftrightarrow [w]_L \subseteq L && \text{per Definition von } F \\ &\Leftrightarrow w \in L && \text{nach Lemma 2.30}\end{aligned}$$

□

## Reguläre Sprachen

---

Im Beweis des Satzes von Myhill und Nerode haben wir gesehen:

1. Wenn  $A = (\Sigma, Q, s, F, \delta)$  ein DEA ist, der die Sprache  $L$  erkennt, und  $u_1, u_2 \in \Sigma^*$  mit  $\delta^*(s, u_1) = \delta^*(s, u_2)$ , dann ist  $[u_1]_L = [u_2]_L$ .

Also ist die Anzahl der  $L$ -Äquivalenzklassen höchstens so groß wie die Anzahl der (erreichbaren) Zustände von  $A$ .

2. Wenn  $L \subseteq \Sigma^*$  nur endlich viele  $L$ -Äquivalenzklassen besitzt, dann gibt es einen DEA, der die Sprache  $L$  erkennt, und der die  $L$ -Äquivalenzklassen als Zustände besitzt.

Also hat dieser DEA—wegen 1.—die *kleinstmögliche* Anzahl von Zuständen unter allen DEAs, die  $L$  erkennen.



## Reguläre Sprachen

---

Einen solchen DEA nennen wir einen *minimalen DEA* (für  $L$ ).

Den speziellen minimalen DEA aus dem Beweis von Satz 2.31 nennen wir den *Myhill-Nerode-Automaten* für  $L$ .

Wir werden später sehen, dass er—in gewissem Sinne—der einzige minimale DEA für  $L$  ist.

Der Beweis von Satz 2.31 zeigt sogar, wie man den Myhill-Nerode-Automaten für  $L$  *konstruiert*, unter der Voraussetzung, dass man die  $L$ -Äquivalenzklassen ‘kennt’, d.h. dass man ihre Anzahl und ihre Elemente kennt (oder zumindest einen Teil der Elemente).

Wir wollen das an einem Beispiel illustrieren.

## Reguläre Sprachen

---

**Beispiel:**  $\Sigma = \{0, 1\}$  und  $L = \{w \in \Sigma^* \mid 0100 \text{ ist Teilwort von } w\}$ .

Bestimmung des Myhill-Nerode-Automaten:

$$[\varepsilon]_L \quad \text{Erg}_L(\varepsilon) = L$$

$$[0]_L \quad \text{Erg}_L(0) = L \cup \{100\} \circ \Sigma^*$$

$$[1]_L = [\varepsilon]_L \quad \text{denn } \text{Erg}_L(1) = L = \text{Erg}_L(\varepsilon)$$

$$\text{also } \delta([\varepsilon]_L, 1) = [1]_L = [\varepsilon]_L$$

$$[00]_L = [0]_L \quad \text{denn } \text{Erg}_L(00) = L \cup \{100\} \circ \Sigma^* = \text{Erg}_L(0)$$

$$\text{also } \delta([0]_L, 0) = [00]_L = [0]_L$$

$$[01]_L \quad \text{Erg}_L(01) = L \cup \{00\} \circ \Sigma^*$$

$$[010]_L \quad \text{Erg}_L(010) = L \cup \{0, 100\} \circ \Sigma^*$$

Man beachte, dass in  $\text{Erg}_L(010)$  nicht nur die Wörter enthalten sind, die mit 0 beginnen, sondern auch die, die mit 100 beginnen, weil ja als letztes eine 0 gelesen wurde.

---

## Reguläre Sprachen

---

Mit anderen Worten: Man muss immer *alle* Präfixe des Suchmusters 0100 beachten, die man gerade gelesen hat, in diesem Falle nicht nur das Präfix 010, sondern auch das Präfix 0.

Weitere  $L$ -Äquivalenzklassen:

$$[011]_L = [\varepsilon]_L \quad \text{denn } \text{Erg}_L(011) = L = \text{Erg}_L(\varepsilon)$$

$$\text{also } \delta([01]_L, 1) = [011]_L = [\varepsilon]_L$$

$$[0100]_L \quad \text{Erg}_L(0100) = \Sigma^*$$

$$[0101]_L = [01]_L \quad \text{denn } \text{Erg}_L(0101) = L \cup \{00\} \circ \Sigma^* = \text{Erg}_L(01)$$

$$\text{also } \delta([010]_L, 1) = [0101]_L = [01]_L$$

$$[0100a]_L = [0100]_L \quad \text{denn } \text{Erg}_L(0100a) = \Sigma^* = \text{Erg}_L(0100)$$

$$\text{also } \delta([0100]_L, a) = [0100a]_L = [0100]_L$$

Damit haben wir alle  $L$ -Äquivalenzklassen und die 'nichttrivialen' Übergänge zwischen ihnen bestimmt.

---

## Reguläre Sprachen

---

Die 'trivialen' Übergänge sind

$$\delta([\varepsilon]_L, 0) = [0]_L$$

$$\delta([0]_L, 1) = [01]_L$$

$$\delta([01]_L, 0) = [010]_L$$

$$\delta([010]_L, 0) = [0100]_L$$

Also sieht der Myhill-Nerode-Automat für  $L$  so aus:

