# Grundlagen der Theoretischen Informatik

## 3. Endliche Automaten (V)

20.05.2015

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

# Organisatorisches

**1. Teilklausur:** Mittwoch, 10.06.2015, D028, 14:30-15:30

(12:00-14:00 - Informationsveranstaltung für Lehramtstudenten)

Anmeldung über MeeToo bis 7.06.2015 um 20:00

`https://userpages.uni-koblenz.de/~metoo/metoo/veranst.php?vnum=2489`

- Wer am 10.06 nicht angemeldet ist, kann an der Teilklausur nicht teilnehmen.
- Wenn Sie an der Teilklausur nicht teilehmen möchten, melden Sie sich bitte ab.

**2. Teilklausur:** Freitag, 7.08.2015, E011, 15:00-16:00

- Teilnahme nur nach bestandener 1. Teilklausur möglich

E-Mail in KLIPS.

**Nachklausur:** 28.09.2015, 14:00-16:00

# Übersicht

- Determinierte endliche Automaten (DEAs)
- Indeterminierte endliche Automaten (NDEAs)
- Automaten mit epsilon-Kanten
- Endliche Automaten akzeptieren genau die Typ-3-Sprachen
- Pumping Lemma
- Abschlusseigenschaften und Wortprobleme
- Rational $=$ Reguläre Ausdrücke

# Pumping-Lemma: Formal

**Theorem (Pumping-Lemma für $L_3$-Sprachen)**

Sei $L \in \mathbf{RAT}$. Dann existiert ein $n \in \mathbb{N}$, so dass:

Für alle

$$x \in L \quad \text{mit} \quad |x| \geq n$$

existiert eine Zerlegung

$$x = uvw \qquad u, v, w \in \Sigma^*$$

mit

- $|v| \geq 1$
- $|v| < n$
- $uv^m w \in L \quad \text{für alle } m \in \mathbb{N}$

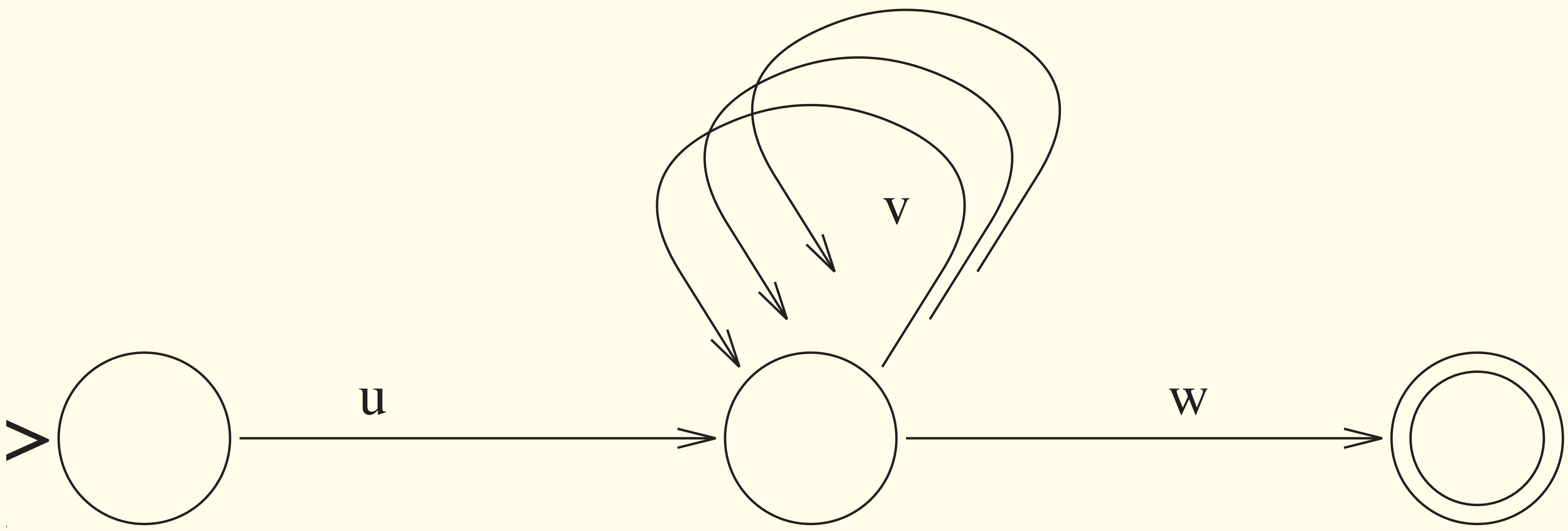
# Pumping-Lemma: Intuition

## Warum gilt das Pumping-Lemma?

- Zu regulärer Sprache $L$ gibt es einen DEA, der $L$ akzeptiert
- Dieser hat endliche Zustandsmenge $K$.
  Sei $m := |K|$.
- Wenn $|w| > |K|$, dann muss beim Akzeptieren von $w$ eine Schleife durchlaufen werden.
- Die Schleife kann auf mehrfach durchlaufen werden.
- Das Teilwort $v$, das der Schleife entspricht kann aufgepumpt werden.

# Pumping-Lemma: Intuition

## Abstrakt gesehen



# Pumping-Lemma: Umkehrung

**Korollar.**
Wenn für eine Sprache das Pumping-Lemma **nicht** gilt,
dann ist sie **nicht** regulär.

**Vorsicht**

- Es gibt nicht-reguläre Sprachen, für die das Pumping-Lemma gilt.
  (Beispiel: $L = \{a^m b^n c^n \mid m, n \geq 1\} \cup \{b^m c^n \mid m, n \geq 0\}$)

- Daraus, dass das Pumping-Lemma für eine Sprache gilt,
  folgt **nicht**, dass sie regulär ist.

# Pumping-Lemma:Anwendung der Umkehrung

**Beispiel**

Folgende Sprachen sind **nicht regulär**:

1. $L_1 := \{a^i b a^i \mid i \in \mathbb{N}\}$

2. $L_2 := \{a^p \mid p \text{ ist Primzahl}\}$

# Pumping-Lemma: Beweis

**Theorem (Pumping-Lemma für $L_3$-Sprachen)**

Sei $L \in \mathbf{RAT}$. Dann existiert ein $n \in \mathbb{N}$, so dass:

Für alle

$$x \in L \quad \text{mit} \quad |x| \geq n$$

existiert eine Zerlegung

$$x = uvw \qquad u, v, w \in \Sigma^*$$

mit

- $|v| \geq 1$
- $|v| < n$
- $uv^m w \in L \quad$ für alle $m \in \mathbb{N}$

# Pumping-Lemma: Beweis

Beweis. Sei $L$ eine reguläre Sprache.

**1. Fall: $L$ ist endlich.**

Sei $w_{max}$ das längste Wort in $L$.

Wir setzen

$$n := |w_{max}| + 1$$

Dann gibt es keine Wörter $x \in L$, für die $|x| \geq n$ gilt.

Also gilt dann Pumping-Lemma trivialerweise.

# Pumping-Lemma: Beweis

Beweis (Fortsetzung)

**2. Fall: $L$ ist unendlich.**

Sei

$$\mathcal{A} = (K, \Sigma, \delta, s_0, F)$$

ein endlicher Automat (DEA), der $L$ akzeptiert.

Wir setzen

$$n := |K| + 1$$

# Pumping-Lemma: Beweis

Beweis (Fortsetzung)

Wir betrachten ein beliebiges Wort

$$x \ = \ x_1 x_2 \dots x_t \ \in \ L \quad \text{mit} \quad |x| = t \geq n, \quad x_i \in \Sigma$$

Zu zeigen: $x$ lässt sich aufpumpen.

Seien

$$q_0, q_1, \dots, q_t \in K,$$

die Zustände, die beim Akzeptieren von $x$ durchlaufen werden:

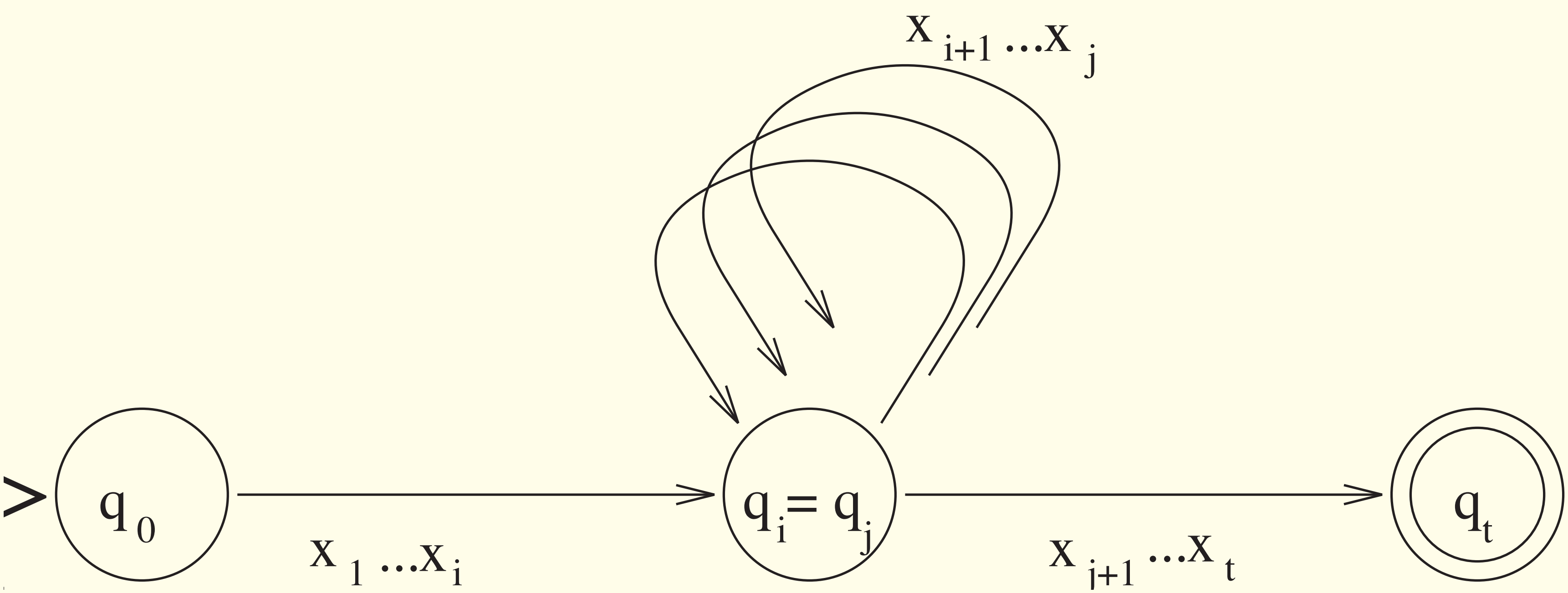
$$q_0 = s_0, \quad q_t \in F, \quad \delta(q_i, x_{i+1}) = q_{i+1} \ \ \forall 0 \leq i \leq t-1$$

# Pumping-Lemma: Beweis

Beweis (Fortsetzung)

Da $t \geq |K| + 1$, muss es $0 \leq i < j \leq t - 1$ geben mit

- $q_i = q_j$
- $|j - i| \leq |K|$



# Pumping-Lemma: Beweis

### Beweis (Fortsetzung)

Wähle nun:

$$\left.\begin{array}{ll} u & x_1 \ldots x_i \\ v & x_{i+1} \ldots x_j \\ w & x_{j+1} \ldots x_t \end{array}\right\} \qquad x = uvw \text{ mit } 1 \leq |v| < n.$$

Damit:

- Für alle $m \geq 0$ gibt es Wege

$$q_0, \ldots, q_{i-1}, \underbrace{q_i, \ldots, q_j = q_i, \ldots, q_j = \;\cdots\; = q_i \ldots, q_j}_{m \text{ mal}}, q_{j+1}, \ldots, q_t$$

- Also: $uv^m w$ wird von $\mathcal{A}$ akzeptiert.
- Also: $uv^m w \in L$ $\square$

# Pumping-Lemma: Stärkere Variante

**Theorem (Pumping-Lemma für L$_3$-Sprachen, stärkere Variante)**
Sei $L \in \textbf{RAT}$. Dann existiert ein $n \in \mathbb{N}$, so dass:

Für alle

$$x \in L \quad \text{mit} \quad |x| \geq n$$

existiert eine Zerlegung

$$x = uvw \qquad u, v, w \in \Sigma^*$$

mit

- $|v| \geq 1$
- $|uv| < n$ (statt $|v| < n$)
- $uv^m w \in L$ für alle $m \in \mathbb{N}$

# Pumping-Lemma: Anwendung der stärkeren Variante

**Beispiel** (Palindrome)

Die Sprache der Palindrome

$$L = \{ww^R \mid w \in \{a, b\}^*\}$$

ist nicht regulär

- Beweis gelingt **nicht** mit der schwächeren Variante des PL (die schwächere Version gilt für die Sprache)
- Beweis **gelingt** mit der stärkeren Varianten des PL

# Pumping-Lemma: Anwendung der stärkeren Variante

**Beispiel** (Palindrome)

Die Sprache der Palindrome

$$L = \{ww^R \mid w \in \{a, b\}^*\}$$

ist nicht regulär

Annahme: $L$ ist regulär. Dann gilt für $L_1$ das Pumping-Lemma. Sei $n$ die Zahl aus dem Pumping-Lemma.

Dann muss sich das Wort $a^n bba^n \in L$ aufpumpen lassen (da $|a^n bba^n| \geq n$).

Sei $a^n bba^n = uvw$ eine passende Zerlegung laut Lemma.

Da $|uv| < n$, ist $u = a^i$, $v = a^j$, $w = a^k bba^n$, wobei $j > 0$ und $i + j + k = n$

Aber dann $uv^0 w = a^{i+k} bba^n \notin L$, da $i + k < n$. Widerspruch.

# Übersicht

- Determinierte endliche Automaten (DEAs)
- Indeterminierte endliche Automaten (NDEAs)
- Automaten mit epsilon-Kanten
- Endliche Automaten akzeptieren genau die Typ-3-Sprachen
- Pumping Lemma
- Abschlusseigenschaften und Wortprobleme
- Rational = Reguläre Ausdrücke

# Abschlusseigenschaften und Wortprobleme

# Abschlusseigenschaften

**Lemma.** Seien zwei reguläre Sprachen $L, L'$ gegeben.

Dann kann man folgende endlichen Automaten konstruieren:

- $\mathcal{A}_\neg$ akzeptiert $\overline{L} = \Sigma^* \setminus L$
- $\mathcal{A}_\cup$ akzeptiert $L \cup L'$
- $\mathcal{A}_\circ$ akzeptiert $L \circ L'$
- $\mathcal{A}_*$ akzeptiert $L^*$
- $\mathcal{A}_\cap$ akzeptiert $L \cap L'$

Beweis: An Tafel.

# Abschlusseigenschaften

Idee:

1) $L = L(\mathcal{A})$ mit $\mathcal{A} = (K, \Sigma, \delta, s_0, F)$

- $\mathcal{A}_{\neg} = (K, \Sigma, \delta, s_0, K \backslash F)$

2) $L = L(\mathcal{A})$, $L' = L(\mathcal{A}')$ mit $\mathcal{A} = (K, \Sigma, \Delta, I, F)$, $\mathcal{A}' = (K', \Sigma', \Delta', I', F')$

- $\mathcal{A}_{\cup} = (K \cup K', \Sigma, \Delta \cup \Delta', I \cup I', F \cup F')$
- $\mathcal{A}_{\circ} = (K \cup K', \Sigma, \Delta \cup \Delta' \cup (F \times \varepsilon) \times I', I, F')$
- $\mathcal{A}_{*} = (K \cup \{\varepsilon_{neu}\}, \Sigma, \Delta \cup (F \times \varepsilon) \times I, I \cup \{\varepsilon_{neu}\}, F \cup \{\varepsilon_{neu}\})$
- $L \cap L' = \overline{(\overline{L} \cup \overline{L'})}$

# Abschlusseigenschaften

**Theorem.** Wenn $L, L'$ reguäre Sprachen sind, dann sind auch

- $\overline{L}$
- $L \cup L'$
- $L \circ L'$
- $L^*$
- $L \cap L'$

reguläre Sprachen.

# Abschlusseigenschaften

**Theorem.** Wenn $L, L'$ reguäre Sprachen sind, dann sind auch

- $\overline{L}$
- $L \cup L'$
- $L \circ L'$
- $L^*$
- $L \cap L'$

reguläre Sprachen.

Beweis.

Gemäß Lemma existieren Automaten, die diese Sprachen akzeptieren.

Also sind sie regulär.

# Anwendung

**Beispiele:**

1. $L_{eq} = \{w \mid \#_a(w) = \#_b(w)\}$ nicht regulär.

# Anwendung

**Beispiele:**

1. $L_{eq} = \{w \mid \#_a(w) = \#_b(w)\}$ nicht regulär.

Beweis. Annahme: $L_{eq}$ regulär

Da $\{a^m c b^n \mid m, n \geq 0\}$ regulär, wäre dann auch

$L_{eq} \cap \{a^m c b^n \mid m, n \geq 0\}$ regulär.

Aber $L_{eq} \cap \{a^m c b^n \mid m, n \geq 0\} = \{a^i c b^i \mid i \geq 0\}$ nicht regulär.

# Anwendung

**Beispiele:**

2. $L_{neq} = \{w \mid \#_a(w) \neq \#_b(w)\}$ nicht regulär.

# Anwendung

**Beispiele:**

2. $L_{neq} = \{w \mid \#_a(w) \neq \#_b(w)\}$ nicht regulär.

Beweis. $L_{eq} = \overline{L_{neq}}$.

Falls $L_{neq}$ regulär wäre, dann wäre auch $L_{eq}$ regulär. Widerspruch.

# Anwendung

**Beispiele:**

3. Die Dycksprache $D_k$ ist nicht regulär

# Anwendung

**Beispiele:**

3. Die Dycksprache $D_k$ ist nicht regulär

**Definition**

Gegeben:

– $k \in \mathbb{N}$

– $\Sigma_k := \{x_1, \overline{x_1}, x_2 \ldots, x_k, \overline{x_k}\}$ ein Alphabet mit $2k$ Symbolen

Die Dycksprache $D_k$ ist die kleinste Menge die folgende Bedingungen erfüllt:

1. $\epsilon \in D_k$,
2. Falls $w \in D_k$, so auch $x_i w \overline{x_i}$.
3. Falls $u$, $v \in D_k$, so auch $uv$.

Interpretiert man die $x_i$ als öffnende, die $\overline{x_i}$ als zugehörige schließende Klammern, so kann man die Dycksprache als die **Menge aller korrekten Klammerausdrücke** sehen.

# Anwendung

**Beispiele:**

3. Die Dycksprache $D_k$ ist nicht regulär

Beweis:

$D_k \cap \{x_i\}^*\{\overline{x_i}\}^* = \{x_i^n\overline{x_i}^n \mid n \in \mathbb{N}\}$.

- $\{x_i\}^*\{\overline{x_i}\}^*$ ist regulär
- $\{x_i^n\overline{x_i}^n \mid n \in \mathbb{N}\}$ ist nicht regulär (Pumping Lemma)