

# Grenzen der Regularität

Um die Mächtigkeit von endlichen Automaten zu verstehen, muss man auch ihre Grenzen kennen.

Sei z.B.  $B = \{0^n 1^n \mid n \geq 0\}$  Gibt es einen DEA für B?

Es sieht so aus, als müsste sich die Maschine die Anzahl der Nullen merken. Nachdem ebendiese Anzahl aber nicht limitiert ist, müsste sich der DEA eine unbeschränkte Anzahl von Möglichkeiten merken können. Mit einer endlichen Anzahl von Zuständen ist das aber nicht möglich!

Reicht das als Beweis?

**NEIN!**

Nur weil es so aussieht, als ob eine Sprache unbegrenzten Speicher brauchen würde, ist das nicht notwendigerweise so!

# Grenzen der Regularität

In der Tat ist  $B = \{ 0^n 1^n \mid n \geq 0 \}$  nicht regulär.

ABER: Andere Sprachen scheinen auch eine unbegrenzte Anzahl von Möglichkeiten zu verlangen, und doch sind sie regulär:

Seien  $C$  und  $D$  Sprachen über  $\Sigma = \{0,1\}$ :

$C = \{ w \mid w \text{ enthält gleich viele Symbole } 0 \text{ wie Symbole } 1 \}$

$D = \{ w \mid \text{die Teilworte } 01 \text{ und } 10 \text{ kommen gleich oft in } w \text{ vor} \}$

Auf den ersten Blick scheint eine akzeptierende Maschine in beiden Fällen unbegrenzt „zählen“ zu müssen, daher könnte man annehmen, dass keine der beiden Sprachen regulär sei.

Wie angenommen, ist  $C$  tatsächlich nicht regulär,  
 $D$  überraschenderweise aber sehr wohl!

# Schubfachprinzip

LEJEUNE DIRICHLET (1805-1859)



Vorlesungen über Zahlentheorie

(prepared for publication by Dedekind, first edition 1863)

Schubfachprinzip:

Bei Verteilung von  $n+1$  Gegenständen auf  $n$  Schubfächer müssen in mindestens einem Schubfach zwei Gegenstände landen. (pigeonhole principle)

# Grenzen der Regularität

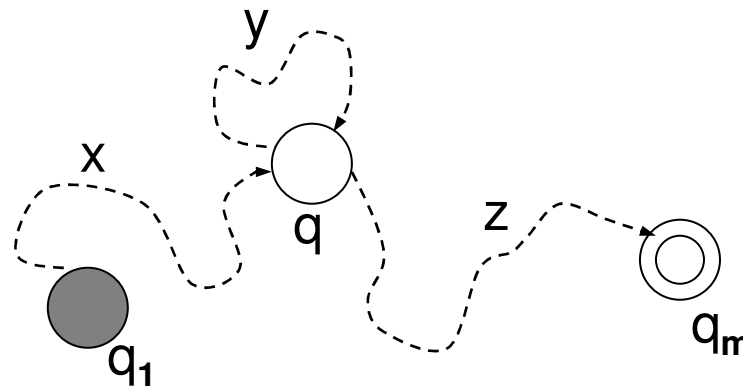
(Unendliche) Reguläre Sprachen haben eine spezielle Eigenschaft:  
Jedes Wort ab einer bestimmten Länge kann „aufgepumpt“ werden,  
d.h. jedes solche Wort enthält ein Teilstück, das beliebig oft wiederholt  
werden kann, wobei die resultierenden Wörter noch immer in  
derselben Sprache liegen.

Deterministischer endlicher Automat habe  $m$  Zustände.

Gilt für ein von  $M$  akzeptiertes Wort  $w$ , dass  $|w| \geq m$ , so muss  
mindestens ein Zustand mehr als einmal durchlaufen werden.

(Schubfachprinzip) Zerlege  $w$  in  $xyz$ :

$y$  gehe von  $q$  nach  $q$  ; diese Schleife kann ausgelassen bzw. beliebig oft  
wiederholt werden, d.h.  $xy^iz$  wird für alle  $i$  ebenfalls von  $M$  akzeptiert !



# Pumping Lemma für reguläre Sprachen A (1)

Sei  $L$  eine unendliche reguläre Sprache. Dann gibt es eine (nur von  $L$  abhängige) Schranke  $m > 0$  so, dass für jedes Wort  $w$  in  $L$  mit  $|w| \geq m$  Wörter  $x, y, z$  so existieren, dass

$$w = xyz \quad \text{mit}$$

$$|y| > 0 \quad \text{und}$$

$$|xy| \leq m$$

sowie

$$w(i) = xy^iz \quad \text{für alle } i \geq 0 \text{ ebenfalls in } L \text{ liegt.}$$

# Pumping Lemma für reguläre Sprachen A (2)

$\forall L \in \mathbf{L}_3$

für jede reguläre Sprache L

$\exists m \in \mathbf{N}$

gibt es eine natürliche Zahl m

$\forall w \in L$  mit  $|w| \geq m$

für jedes Wort w aus L (mind. m lang)

$\exists xyz = w$

gibt es Teilwörter x,y,z von w so, dass

$|y| > 0$

das Teilwort y nicht leer ist,

$|xy| \leq m$

die ersten zwei Teilwörter sind nicht länger als m sowie

$\forall i \in \mathbf{N}$

für jede natürliche Zahl i

$xy^iz \in L$

ist die i-fach gepumpte Version in L.

# Pumping Lemma für reguläre Sprachen A (3)

$$L \notin \mathcal{L}_3$$

für nicht-reguläre Sprachen L

┐

$$\exists m \in \mathbf{N}$$

gibt es eine natürliche Zahl m

$$\forall w \in L \text{ mit } |w| \geq m$$

für jedes Wort w aus L (mind. m lang)

$$\exists xyz = w$$

gibt es Teilwörter x,y,z von w so, dass

$$|y| > 0$$

das Teilwort y nicht leer ist,

$$|xy| \leq m$$

die ersten zwei Teilwörter sind nicht länger als m sowie

$$\forall i \in \mathbf{N}$$

für jede natürliche Zahl i

$$xy^iz \in L$$

ist die i-fach gepumpte Version in L.

# Pumping Lemma für reguläre Sprachen A (4)

$$L \notin L_3$$

für nicht-reguläre Sprachen  $L$

$$\forall m \in \mathbf{N}$$

für alle natürlichen Zahl  $m$

$$\exists w \in L \text{ mit } |w| \geq m$$

existiert ein Wort  $w$  aus  $L$  (mind.  $m$  lang)

$$\forall x,y,z \text{ mit } xyz = w$$

für alle Zerlegungen von  $w$  in  $xyz$ , wo

$$|y| > 0$$

das Teilwort  $y$  nicht leer ist und

$$|xy| \leq m$$

die ersten zwei Teilwörter  
nicht länger als  $m$  sind,

$$\exists i \in \mathbf{N}$$

existiert eine natürliche Zahl  $i$  so, dass

$$xy^iz \notin L$$

die  $i$ -fach gepumpte Version nicht in  $L$ .



# Pumping Lemma - Beispiel

$$L = \{ 0^n 1^n \mid n \geq 0 \}$$

Beweis durch Widerspruch:

Angenommen  $L$  sei regulär.

Für beliebiges  $m$  (Konstante aus dem Pumping Lemma) wähle  $w = 0^m 1^m$  (offensichtlich gilt  $w \in L$  und  $|w| \geq m$ ).

Betrachte beliebige Zerlegungen von  $w$  in  $xyz$  mit  $|xy| \leq m$  und  $|y| > 0$ :  $xy$  kann nur aus Nullen bestehen.

Wähle ein  $i$  so, dass  $xy^i z$  nicht von der Gestalt  $0^n 1^n$  ist:

Für alle  $i$  gilt, dass  $xy^i z = 0^{m+(i-1)|y|} 1^m$  ist, d.h. für

$i = 0$  und alle  $i > 1$  ist  $m+(i-1)|y| \neq m$  und

$xy^i z$  daher nicht von der Gestalt  $0^n 1^n$  !



# Pumping Lemma für reguläre Sprachen B

Sei  $L$  eine unendliche reguläre Sprache. Dann gibt es eine (nur von  $L$  abhängige) Schranke  $m > 0$  so, dass für jedes Wort  $w$  in  $L$  mit  $|w| \geq m$  und jedes Teilwort  $v$  von  $w$  mit  $|v| \geq m$  und  $w = svt$  für Wörter  $s, t \in \Sigma^*$  gilt:

$$v = xyz \quad \text{mit}$$

$$|xy| \leq m \text{ und}$$

$$|y| > 0$$

sowie

$$w(i) = sxy^izt \quad \text{für alle } i \geq 0 \text{ ebenfalls in } L \text{ liegt.}$$

Diese zweite Variante des Pumping Lemmas ist eine Verallgemeinerung der ersten Variante, bei der ein beliebiges Teilwort ausreichender Länge ausgewählt werden kann, von dem nun wieder ein Teilwort beschränkter Länge beliebig aufgepumpt werden kann.

# Pumping Lemma für reguläre Sprachen - Beispiele

Zeigen Sie mit einer der beiden Variante des Pumping Lemmas für reguläre Sprachen, dass die folgenden formalen Sprachen nicht regulär sind.

Aufgabe PLR A:

$$L = \{ 0^n 1^m \mid m \geq n \}$$

Aufgabe PLR B:

$$L = \{ 0^n 1^m \mid n \geq m \}$$

# Homomorphismus

Seien  $\Sigma$  und  $\Gamma$  zwei Alphabete.

Ein **Homomorphismus** ist eine Funktion  $h: \Sigma \rightarrow \Gamma^*$ .

Erweiterung auf Wörter über  $\Sigma$ :

$$h(s_1 \dots s_n) = h(s_1) \dots h(s_n),$$

wobei  $s_1, \dots, s_n$  Symbole aus  $\Sigma$  sind.

Homomorphes Bild einer Sprache  $L$ :

$$h(L) = \{ h(w) \mid w \in L \}$$

# Homomorphismus – Beispiel 1

Sei  $L = \{ (a^3b^3)^n a^k (c^7d^8)^n \mid n \geq 0 \}$ .

$h: \{a,b,c,d\}^* \rightarrow \{a,b\}^*$

$h(a) = \varepsilon, h(b) = a, h(c) = b, h(d) = \varepsilon$ .

Homomorphes Bild der Sprache L:

$h(L) = \{ a^{3n}b^{7n} \mid n \geq 0 \}$

Ganz allgemein: Die Sprachen  $\{ a^{kn}b^{mn} \mid n \geq 0 \}$ ,  $k, m > 0$ , sind nicht regulär.

Die Familie der regulären Sprachen ist gegenüber Homomorphismen abgeschlossen, daher kann auch L nicht regulär sein.

# Homomorphismus – Beispiel 2

Sei  $\Sigma$  ein Alphabet,

$$L = \{ w \in \Sigma^* \mid w = v v^R \text{ für ein } v \in \Sigma^* \}.$$

$v^R$  ... Spiegelbild von  $v \in \Sigma^*$

$$v = s_1 \dots s_n, v^R = s_n \dots s_1,$$

wobei  $s_1, \dots, s_n$  Symbole aus  $\Sigma$  sind.

$$h: \Sigma \rightarrow \{a\}^*.$$

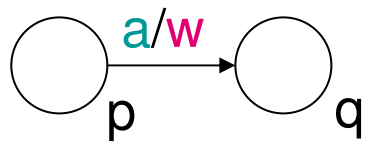
Homomorphes Bild der Sprache  $L$ :

$$h(L) = \{ a^{2^n} \mid n \geq 0 \} \quad (\text{reguläre Sprache!})$$

# Verallgemeinerte Sequentielle Maschinen

generalized sequential machines

Endliche Automaten mit Ausgabe, d.h. mit jedem eingelesenen Symbol wird ein Wort ausgegeben. Die bei der erfolgreichen Analyse von Eingabewörtern aus einer Sprache  $L$  erzeugten Ausgabewörter bilden die durch die gsm-Abbildung erzeugte Sprache.

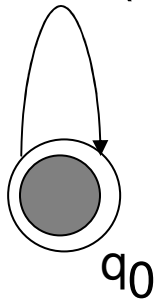


Beim Übergang vom Zustand  $p$  in den Zustand  $q$  wird ein Symbol  $a$  gelesen und das Wort  $w$  ausgegeben.

# gsm – Beispiel 1

Jeder Homomorphismus  $h$  ist eine gsm-Abbildung:

$a/h(a)$  für alle  $a$  aus  $T$



$$M = (\{q_0\}, T, T', \delta, q_0, \{q_0\})$$

$$\delta(q_0, 0) = (q_0, h(a)) \text{ für alle } a \text{ aus } T$$

Die Familie der regulären Sprachen ist abgeschlossen gegenüber gsm-Abbildungen.

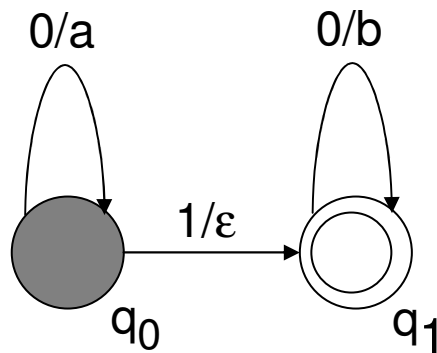
Aus dem obigen Beispiel folgt sofort auch:

Die Familie der regulären Sprachen ist abgeschlossen gegenüber Homomorphismen.



## gsm – Beispiel 2

**gsm M** bildet  $\{0^n 1 0^n \mid n \geq 0\}$  auf  $\{a^n b^n \mid n \geq 0\}$  ab.



$$M = (\{q_0, q_1\}, \{0, 1\}, \{a, b\}, \delta, q_0, \{q_1\})$$

$$\delta(q_0, 0) = (q_0, a)$$

$$\delta(q_0, 1) = (q_1, \varepsilon)$$

$$\delta(q_1, 0) = (q_1, b)$$

Die Familie der regulären Sprachen ist abgeschlossen gegenüber gsm-Abbildungen, daher kann auch  $L$  nicht regulär sein, da  $\{a^n b^n \mid n \geq 0\}$  nicht regulär ist.

## gsm - Aufgaben

Konstruieren Sie eine **gsm** **M**, welche

A)  $\{0^{2n}10^{3n} \mid n \geq 1\}$  auf  $\{a^n b^n \mid n \geq 1\}$ ,

B)  $\{0^{2n}10^{3n}10^{4n} \mid n \geq 0\}$  auf  $\{a^{4n}b^{3n-1}c^{64n+16} \mid n \geq 0\}$ ,

C)  $\{(a^3b^5)^{2n}c^{3n} \mid n \geq 0\}$  auf  $\{a^{16n}b^{81n} \mid n \geq 1\}$ ,

abbildet.

# Komplement regulärer Sprachen

$L_{\text{reg}}(\Sigma)$  ist abgeschlossen gegenüber Komplement bzgl.  $\Sigma^*$

Konstruiere DEA  $M = (Q, \Sigma, \delta, q_0, F)$  für  $L$ .

Konstruiere daraus DEA  $M'$  für  $\Sigma^* - L$ :

vertausche End- und Nichtendzustände

$$M' = (Q, \Sigma, \delta, q_0, Q - F)$$

Wichtig:  $Q$  enthält eventuelle Falle, d.h.,  
Vollständigkeit der Übergangsfunktion ist wichtig!

# Eigenschaften regulärer Sprachen

$L_{\text{reg}}(\Sigma)$  ist abgeschlossen gegenüber

- Vereinigung
- Konkatenation
- Stern
- Plus-Operator  $(A^+ = AA^*)$
- Komplement bzgl.  $\Sigma^*$
- Durchschnitt  $(A \cap B = (A^c \cup B^c)^c)$
- Differenz  $(A - B = A \cap B^c)$
- Homomorphismen
- gsm-Abbildungen

$A^c$  bezeichnet das Komplement der formalen Sprache  $A$  bezüglich  $\Sigma^*$