



# Informatik III

## *WS 03/04*

Prof. Dr. Dorothea Wagner

`dwagner@ira.uka.de`

Kapitel 2 : Endliche Automaten

## 2.1 Definition

*Ein (deterministischer) endlicher Automat (D)EA besteht aus:*

- $Q$ , einer endlichen Menge von **Zuständen**;
- $\Sigma$ , einer endlichen Menge von **Eingabesymbolen**,  
**Alphabet**;
- $\delta: Q \times \Sigma \rightarrow Q$ , einer **Übergangsfunktion**;
- $s \in Q$ , einem **Startzustand**;
- $F \subseteq Q$ , einer Menge von **Endzuständen**.

## 2.2 Definition

- Ein endliches **Alphabet**  $\Sigma$  ist eine endliche Menge von Symbolen.
- Eine endliche Folge von Symbolen aus  $\Sigma$  heißt **Wort** (über  $\Sigma$ ).
- Die Menge aller Wörter über  $\Sigma$  heißt  $\Sigma^*$ .
- Die Anzahl der Symbole eines Wortes  $w$  ist die **Länge** von  $w$ , sie wird durch die Kardinalität von  $w$  ( $|w|$ ) bezeichnet.
- Das **leere Wort** heißt  $\varepsilon$  ( $|\varepsilon| = 0$ ); es gilt  $\varepsilon \in \Sigma^*$  f. a.  $\Sigma$ .

# Deterministische endl. Automaten u. formale Sprachen

- Aus zwei Wörtern  $w_1, w_2$  erhält man die **Konkatenation**, d.h. ein Wort  $w = w_1 \cdot w_2$ , durch Hintereinanderschreiben.

$$w^i := \underbrace{w \cdot \dots \cdot w}_{i\text{-Mal}} \quad w^0 := \varepsilon$$

Oft schreiben wir statt  $w_1 \cdot w_2$  auch nur  $w_1w_2$ .

## 2.3 Definition

*Eine Menge  $L$  von Wörtern über einem Alphabet  $\Sigma$ ,  
d.h.  $L \subseteq \Sigma^*$ , heißt **(formale) Sprache** über  $\Sigma$ .*

## 2.4 Definition

*Läßt sich ein Wort  $w$  schreiben als  $w = u \cdot v \cdot x$ , wobei  $u, v, x$  beliebige Wörter sind, so heißt:*

$u$	Präfix	} von $w$
$v$	Teilwort	
$x$	Suffix	

## 2.5 Definition

Seien  $L, L_1, L_2 \subseteq \Sigma^*$  Sprachen.

**Produktsprache:**

$$L_1 \cdot L_2 := \{w_1 \cdot w_2 \mid w_1 \in L_1, w_2 \in L_2\}$$

**$k$ -faches Produkt:**

$$L^k := \{w_1 \cdot \dots \cdot w_k \mid w_i \in L, 1 \leq i \leq k\}$$
$$L^0 := \{\varepsilon\}$$

**Quotientensprache:**

$$L_1 / L_2 := \{w \in \Sigma^* \mid \exists z \in L_2 \text{ mit } w \cdot z \in L_1\}$$

# Deterministische endl. Automaten u. formale Sprachen

**Kleene'scher Abschluß:**

$$L^* := \bigcup_{i \geq 0} L^i = \{w_1 \cdot \dots \cdot w_n \mid w_i \in L, n \in \mathbb{N}_0\}$$

**positiver Abschluß:**  $L^+ := \bigcup_{i > 0} L^i$

**Komplementsprache:**  $L^c := \Sigma^* \setminus L$

## 2.6 Definition

- Ein endlicher Automat **erkennt** oder **akzeptiert** eine Sprache  $L$ , d.h. eine Menge von Wörtern über dem Alphabet des Automaten, wenn er nach Abarbeitung eines Wortes  $w$  genau dann in einem Endzustand ist, wenn das Wort  $w$  in der Sprache  $L$  ist ( $w \in L$ ).
- Eine formale Sprache heißt **endliche Automatensprache**, wenn es einen endlichen Automaten gibt, der sie erkennt.

## 2.7 Definition

Eine Sprache  $L \subseteq \Sigma^*$  heißt **regulär**, wenn für sie einer der folgenden Punkte gilt: (induktive Definition)

I. Verankerung:

(1)  $L = \{a\}$  mit  $a \in \Sigma$  oder

(2)  $L = \emptyset$

II. Induktion: Seien  $L_1, L_2$  reguläre Sprachen

(3)  $L = L_1 \cdot L_2$  oder

(4)  $L = L_1 \cup L_2$  oder

(5)  $L = L_1^*$

## 2.8 Definition

Sei  $\Sigma$  eine Alphabet. Eine reguläre Sprache über  $\Sigma$  kann durch einen **regulären Ausdruck** beschrieben werden.

Dabei bezeichnet:

- $\emptyset$  den regulären Ausdruck, der die leere Menge beschreibt.
- $\varepsilon$  den regulären Ausdruck, der die Menge  $\{\varepsilon\}$  beschreibt.
- $a$  den regulären Ausdruck, der die Menge  $\{a\}$  beschreibt.

## 2.9 Satz

*Jede reguläre Sprache wird von einem (deterministischen) endlichen Automaten (DEA) akzeptiert.*

# Nichtdeterministische endliche Automaten

## 2.10 Definition

1. Ein **nichtdeterministischer endlicher Automat (NEA)** besteht aus:

- $Q$ , einer endlichen Zustandsmenge;
- $\Sigma$ , einem endlichen Alphabet;
- $\delta$ , einer Übergangsfunktion  
 $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$ , wobei  $2^Q$  die Potenzmenge von  $Q$  darstellt;
- $s$ , einem Startzustand;
- $F$ , einer Menge von Endzuständen.

# Nichtdeterministische endliche Automaten

2. Ein nichtdeterministischer endlicher Automat **akzeptiert** ein Wort  $w \in \Sigma^*$ , wenn es eine Folge von Übergängen gibt (auch  $\varepsilon$ -Übergänge), so daß er bei Eingabe von  $w$  in einen Endzustand gelangt, d.h. bei Eingabe von  $w$  ein Endzustand erreichbar ist.

# Nichtdeterministische endliche Automaten

## **Bemerkung:**

*Bei der Abarbeitung eines Eingabesymbols aus  $\Sigma$  kann der Automat sich — nichtdeterministisch — aussuchen, in welchen Zustand aus einer Teilmenge von  $Q$  er geht. Er kann auch ohne Lesen eines Eingabesymbols „spontan“ sogenannte  $\varepsilon$ -Übergänge ausführen.  $\delta(q, a)$  kann auch  $\emptyset$  sein, d.h. es gibt zu  $q$  bei Lesen von  $a$  keinen Folgezustand.*

# Nichtdeterministische endliche Automaten

## 2.11 Definition

*Zwei endliche Automaten, die dieselbe Sprache akzeptieren, heißen **äquivalent**.*

## 2.12 Satz (Äquivalenz von NEA's und DEA's)

*Zu jedem nichtdeterministischen endlichen Automaten gibt es einen äquivalenten deterministischen endlichen Automaten.*

# Nichtdeterministische endliche Automaten

## 2.13 Satz

*Zu jedem nichtdeterministischen endlichen Automat mit  $\varepsilon$ -Übergängen gibt es einen äquivalenten nichtdeterministischen endlichen Automaten ohne  $\varepsilon$ -Übergängen, der nicht mehr Zustände hat.*

# Nichtdeterministische endliche Automaten

## Umkehrung von Satz 2.9

### 2.14 Satz

*Jede Sprache, die von einem endlichen Automaten erkannt wird, ist regulär.*

# Nichtdeterministische endliche Automaten

## 2.15 Satz (Pumping–Lemma für reguläre Sprachen)

*Sei  $L$  eine reguläre Sprache. Dann existiert eine Zahl  $n \in \mathbb{N}$ , so dass für jedes Wort  $w \in L$  mit  $|w| > n$  eine Darstellung*

$$w = uvx \text{ mit } |uv| \leq n, v \neq \varepsilon,$$

*existiert, bei der auch  $uv^i x \in L$  ist für alle  $i \in \mathbb{N}_0$ .*

# Nichtdeterministische endliche Automaten

## 2.16 Satz (Verallgem. Pumping Lemma für reguläre Sprachen)

*Sei  $L$  eine reguläre Sprache. Dann existiert eine Zahl*

*$n \in \mathbb{N}$ , so dass für jedes Wort  $w \in L$  mit  $|w| \geq n$  und jede Darstellung  $w = tyx$  mit  $|y| = n$  gilt:*

*für das Teilwort  $y$  existiert eine Darstellung  $y = uvz$  mit  $v \neq \varepsilon$  bei der auch  $tuv^i zx \in L$  ist für alle  $i \in \mathbb{N}_0$ .*

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.17 Definition

*Zustände eines (deterministischen) endlichen Automaten, die vom Anfangszustand aus nicht erreichbar sind, heißen **überflüssig**.*

## 2.18 Satz

*Die Menge aller überflüssigen Zustände eines (deterministischen) endlichen Automaten kann in der Zeit  $\mathcal{O}(|Q| \cdot |\Sigma|)$  berechnet werden.*

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.19 Definition

Zwei Zustände  $p$  und  $q$  eines deterministischen endlichen Automaten heißen **äquivalent** ( $p \equiv q$ ), wenn für alle Wörter  $w \in \Sigma^*$  gilt:

$$\delta(p, w) \in F \iff \delta(q, w) \in F.$$

Offensichtlich ist  $\equiv$  eine Äquivalenzrelation. Mit  $[p]$  bezeichnen wir die Äquivalenzklasse der zu  $p$  äquivalenten Zustände.

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.20 Definition

Zu einem deterministischen endlichen Automaten

$\mathcal{A} = (Q, \Sigma, \delta, s, F)$  definieren wir den  
Äquivalenzklassenautomaten

$\mathcal{A}^{\equiv} = (Q^{\equiv}, \Sigma^{\equiv}, \delta^{\equiv}, s^{\equiv}, F^{\equiv})$  durch:

- $Q^{\equiv} := \{[q] \mid q \in Q\}$
- $\Sigma^{\equiv} := \Sigma$
- $\delta^{\equiv}([q], a) := [\delta(q, a)]$
- $s^{\equiv} := [s]$
- $F^{\equiv} := \{[f] \mid f \in F\}$

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.21 Satz

*Der Äquivalenzklassenautomat  $\mathcal{A}^{\equiv}$  zu einem deterministischen endlichen Automaten  $\mathcal{A}$  ist wohldefiniert.*

## 2.22 Satz

*Der Äquivalenzklassenautomat  $\mathcal{A}^{\equiv}$  zu  $\mathcal{A}$  akzeptiert dieselbe Sprache wie  $\mathcal{A}$ .*

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.23 Definition

Eine Äquivalenzrelation  $R$  über  $\Sigma^*$  heißt **rechtsinvariant**, wenn für alle  $x, y \in \Sigma^*$  gilt:

*falls  $x R y$  so gilt auch  $xz R yz$  für alle  $z \in \Sigma^*$ .*

Den **Index** von  $R$  bezeichnen wir mit  **$\text{ind}(R)$** ; er ist die Anzahl der Äquivalenzklassen von  $\Sigma^*$  bezüglich  $R$ .

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.24 Definition

Für eine Sprache  $L \subseteq \Sigma^*$  ist die **Nerode-Relation**  $R_L$  definiert durch:

für  $x, y \in \Sigma^*$  ist  $x R_L y$  genau dann wenn  
 $(xz \in L \Leftrightarrow yz \in L)$  für alle  $z \in \Sigma^*$  gilt.

## 2.25 Satz (von Nerode)

*Die folgenden Aussagen sind äquivalent:*

- 1.  $L \subseteq \Sigma^*$  wird von einem deterministischen endlichen Automaten akzeptiert.*
- 2.  $L$  ist die Vereinigung von Äquivalenzklassen einer rechtsinvarianten Äquivalenzrelation mit endlichem Index.*
- 3. Die Nerode–Relation hat endlichen Index.*

# Minimierung v. Automaten, Äquivalenzklassenautomat

## 2.26 Korollar

*Der im dritten Beweisteil zum Satz von Nerode konstruierte Automat  $\mathcal{A}$  zu  $R_L$  — der **Automat der Nerode-Relation** — ist minimal.*

## 2.27 Satz

*Der Äquivalenzklassenautomat  $A^{\equiv}$  zu einem deterministischen endlichen Automaten  $\mathcal{A}$  ohne überflüssige Zustände ist minimal.*