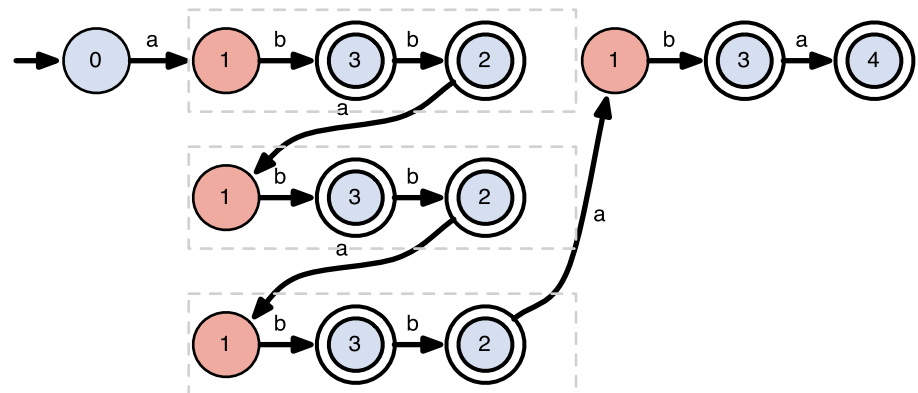


Berechenbarkeitstheorie

8. Vorlesung



Dr. Franziska Jahnke

Institut für Mathematische Logik und Grundlagenforschung

WWU Münster

Kontextfreie Grammatiken

2

Kontextfreie Grammatiken

2

- Grammatiken = Formalismus zum Beschreiben von Sprachen mit Ableitungsregeln

Kontextfreie Grammatiken

2

- Grammatiken = Formalismus zum Beschreiben von Sprachen mit Ableitungsregeln
- Ziel: $\{a^n b^n \mid n \geq 0\}$ soll erkannt werden

Kontextfreie Grammatiken

2

- Grammatiken = Formalismus zum Beschreiben von Sprachen mit Ableitungsregeln
- Ziel: $\{a^n b^n \mid n \geq 0\}$ soll erkannt werden
- Ziel: Modell soll *beherrschbar* bleiben

- Grammatiken = Formalismus zum Beschreiben von Sprachen mit Ableitungsregeln
- Ziel: $\{a^n b^n \mid n \geq 0\}$ soll erkannt werden
- Ziel: Modell soll *beherrschbar* bleiben

Kontextfreie Grammatik

Eine Kontextfreie Grammatik G ist ein 4 Tupel

(V, Σ, R, S) , bestehend aus

- V endliche Menge von **Variablen**,
- $\Sigma \neq \emptyset$ endliche Menge von **Terminalsymbolen**,
- $R \subseteq V \times (V \cup \Sigma)^*$ einer endliche Menge von **Regeln**,
- $S \in V$ einem **Startsymbol**.

Sprache einer kf Grammatik

3

Sprache einer kf Grammatik

3

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab

Sprache einer kf Grammatik

3

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab

Schreibweise: $uAv \Rightarrow uww$

Sprache einer kf Grammatik

3

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab

Schreibweise: $uAv \Rightarrow uww$

- wiederholtes Ableiten: $uAv \Rightarrow \dots \Rightarrow \dots \Rightarrow t$

Sprache einer kf Grammatik

3

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab

Schreibweise: $uAv \Rightarrow uww$

- wiederholtes Ableiten: $uAv \Rightarrow \dots \Rightarrow \dots \Rightarrow t$

Schreibweise: $uAv \Rightarrow^* t$

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab
Schreibweise: $uAv \Rightarrow uww$
- wiederholtes Ableiten: $uAv \Rightarrow \dots \Rightarrow \dots \Rightarrow t$
Schreibweise: $uAv \Rightarrow^* t$

Definition

Für eine kf Grammatik G ist die Sprache der Grammatik $L(G)$ definiert als

$$L(G) := \{w \in \Sigma^* \mid S \Rightarrow^* w\}.$$

- wenn $u, v, w \in (\Sigma \cup V)^*$ und $A \rightarrow w$ eine Regel, dann leitet uAv zu uww ab
Schreibweise: $uAv \Rightarrow uww$
- wiederholtes Ableiten: $uAv \Rightarrow \dots \Rightarrow \dots \Rightarrow t$
Schreibweise: $uAv \Rightarrow^* t$

Definition

Für eine kf Grammatik G ist die Sprache der Grammatik $L(G)$ definiert als

$$L(G) := \{w \in \Sigma^* \mid S \Rightarrow^* w\}.$$

- also alle Wörter **über** Σ die ich aus dem Startsymbol ableiten kann

Definition

Die Sprache $L(G)$ einer kf. Grammatik G heißt **kontextfreie Sprache**.

$$\text{CFL} := \{L \mid L \text{ ist kontextfrei}\}$$

Definition

Die Sprache $L(G)$ einer kf. Grammatik G heißt **kontextfreie Sprache**.

$$\text{CFL} := \{L \mid L \text{ ist kontextfrei}\}$$

Satz 10

$$L \in \text{REG} \Rightarrow L \in \text{CFL}$$

Definition

Die Sprache $L(G)$ einer kf. Grammatik G heißt **kontextfreie Sprache**.

$$\text{CFL} := \{L \mid L \text{ ist kontextfrei}\}$$

Satz 10

$$L \in \text{REG} \Rightarrow L \in \text{CFL}$$

- Aus Satz 10 folgt, dass $\text{REG} \subsetneq \text{CFL} \dots$

Definition

Die Sprache $L(G)$ einer kf. Grammatik G heißt **kontextfreie Sprache**.

$$\text{CFL} := \{L \mid L \text{ ist kontextfrei}\}$$

Satz 10

$$L \in \text{REG} \Rightarrow L \in \text{CFL}$$

- Aus Satz 10 folgt, dass $\text{REG} \subsetneq \text{CFL} \dots$
... denn $\{a^n b^n \mid n \geq 0\}$ ist kontextfrei aber nicht regulär

Definition

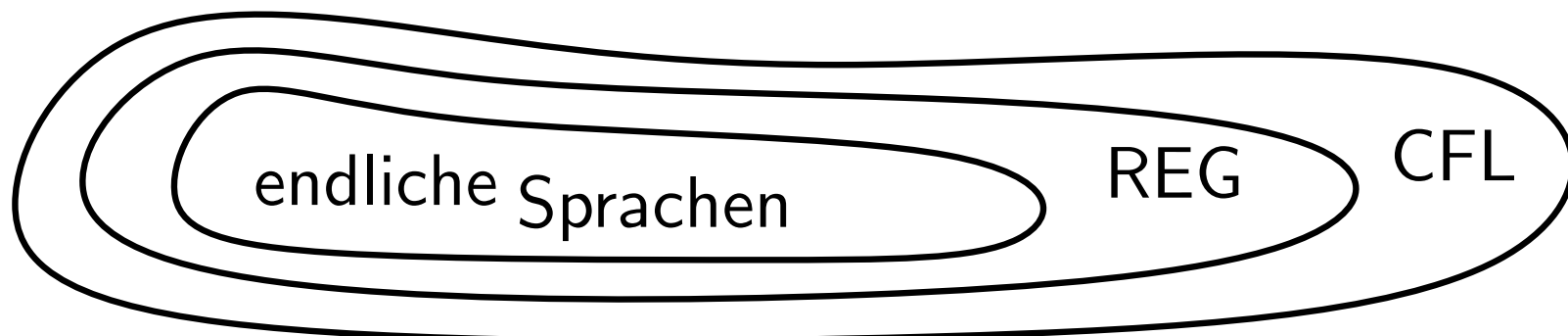
Die Sprache $L(G)$ einer kf. Grammatik G heißt **kontextfreie Sprache**.

$$\text{CFL} := \{L \mid L \text{ ist kontextfrei}\}$$

Satz 10

$$L \in \text{REG} \Rightarrow L \in \text{CFL}$$

- Aus Satz 10 folgt, dass $\text{REG} \subsetneq \text{CFL} \dots$
... denn $\{a^n b^n \mid n \geq 0\}$ ist kontextfrei aber nicht regulär



Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

5

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

5

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

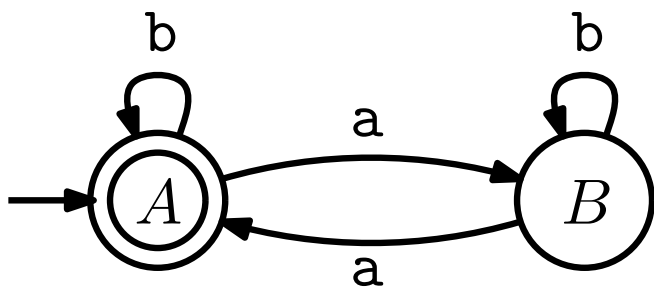
- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

Konstruktion am Beispiel

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

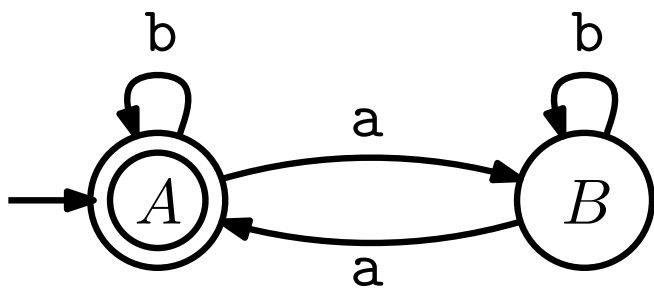
Konstruktion am Beispiel



Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

Konstruktion am Beispiel

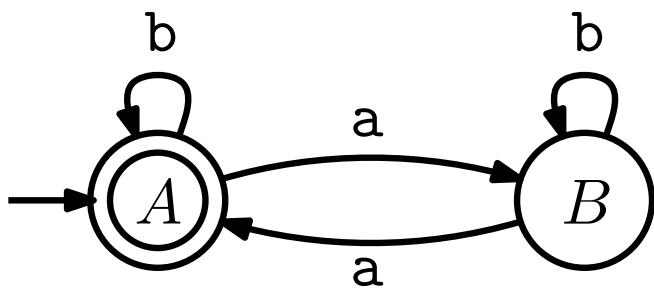


$$V = \{A, B\}, \Sigma = \{a, b\}, S = A$$

Beweis $L \in \text{REG} \Rightarrow L \in \text{CFL}$

- sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, welcher L erkennt
- Definiere Grammatik $G_M = (V, \Sigma, R, S)$ mit
 - $V = Q$,
 - $S = q_0$,
 - für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

Konstruktion am Beispiel



$$V = \{A, B\}, \Sigma = \{a, b\}, S = A$$

$$R: \begin{array}{l} A \rightarrow aB \mid bA \mid \varepsilon \\ B \rightarrow aA \mid bB \end{array}$$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
 - für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.
-

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1 x_2 \dots$ ↘ (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.


■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1 x_2 \dots$

↘ (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1} p_{i+1}$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$


 (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L$ \iff w -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$ \implies (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

$\iff w \in L(G_M)$

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L$ \iff w -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$

\curvearrowright (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

$\iff w \in L(G_M)$

□

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$

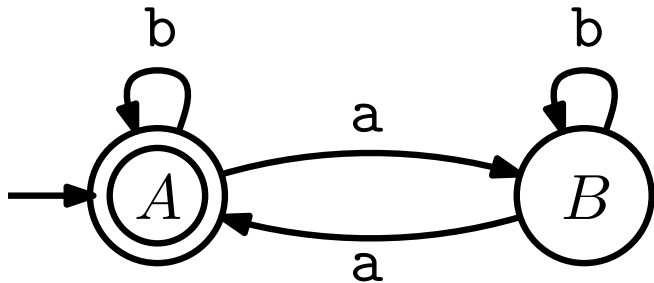
↘ (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

$\iff w \in L(G_M)$

Bsp.

aba hat akz. Lauf (A, B, B, A)

□

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$

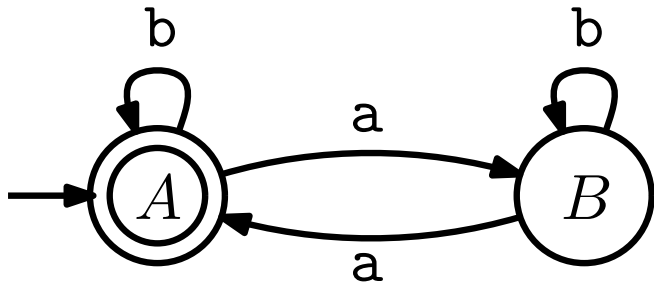
↘ (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

$\iff w \in L(G_M)$

Bsp.



$$R: \begin{array}{l} A \rightarrow aB \mid bA \mid \varepsilon \\ B \rightarrow aA \mid bB \end{array}$$

aba hat akz. Lauf (A, B, B, A)

□

- für alle $q \in Q$ und $a \in \Sigma$ mit $\delta(q, a) = p$ füge die Regel $q \rightarrow ap$ zu R hinzu,
- für alle $q \in F$ füge die Regel $q \rightarrow \varepsilon$ zu R hinzu.

■ $w \in L \iff w$ -Lauf (p_0, p_1, \dots, p_n) in M und $p_n \in F$
 $w = x_1x_2 \dots$

↘ (i) $p_0 = q_0$ (ii) $\delta(p_i, x_{i+1}) = p_{i+1}$

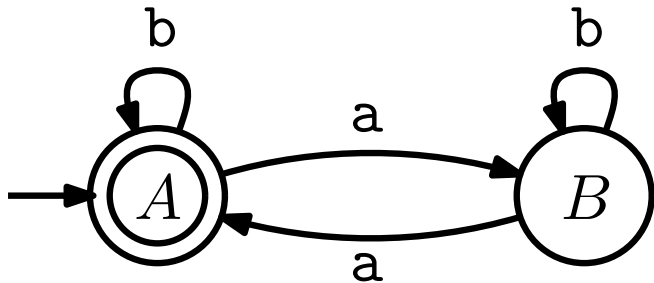
$\iff (p_0, p_1, \dots, p_n) \in V^n$ mit $p_n \rightarrow \varepsilon$
 $p_0 = S$ und $p_i \rightarrow x_{i+1}p_{i+1}$

$\iff p_0 \Rightarrow x_1p_1 \Rightarrow x_1x_2p_2 \Rightarrow \dots \Rightarrow wp_n \Rightarrow w$

$\iff w \in L(G_M)$

□

Bsp.



$$R: \begin{array}{l} A \rightarrow aB \mid bA \mid \varepsilon \\ B \rightarrow aA \mid bB \end{array}$$

aba hat akz. Lauf (A, B, B, A)

$$A \Rightarrow aB \Rightarrow abB \Rightarrow abaA \Rightarrow aba$$

Chomsky Normalform

7

8. Vorlesung

Definition

Eine kontextfreie Grammatik G ist in **Chomsky Normalform (CNF)** wenn alle ihre Regeln von der folgenden Form sind:

1. $A \rightarrow BC$, wobei $B, C \in V$ und $B, C \neq S$
2. $A \rightarrow a$, wobei $a \in \Sigma$
3. $S \rightarrow \varepsilon$

Definition

Eine kontextfreie Grammatik G ist in **Chomsky Normalform (CNF)** wenn alle ihre Regeln von der folgenden Form sind:

1. $A \rightarrow BC$, wobei $B, C \in V$ und $B, C \neq S$
2. $A \rightarrow a$, wobei $a \in \Sigma$
3. $S \rightarrow \varepsilon$

Bsp.

$$\begin{array}{l} S \rightarrow BB \mid AA \mid \varepsilon \\ A \rightarrow AB \mid BA \mid AA \mid a \\ B \rightarrow b \end{array}$$

Definition

Eine kontextfreie Grammatik G ist in **Chomsky Normalform (CNF)** wenn alle ihre Regeln von der folgenden Form sind:

1. $A \rightarrow BC$, wobei $B, C \in V$ und $B, C \neq S$
2. $A \rightarrow a$, wobei $a \in \Sigma$
3. $S \rightarrow \varepsilon$

Bsp.

$$\begin{array}{l} S \rightarrow BB \mid AA \mid \varepsilon \\ A \rightarrow AB \mid BA \mid AA \mid a \\ B \rightarrow b \end{array}$$

Satz 11

Für jede Sprache $L \in \text{CFL}$ gibt es eine kf. Grammatik G in CNF mit $L(G) = L$.

Beweis

- Sei $G = (V, \Sigma, R, S)$ eine kfG für L , dann konstruieren wir eine kfG $G' = (V', \Sigma, R', S')$, in CNF für L
- Konstruktion erfolgt in 5 Schritten durch Umbau der Regeln aus R

Beweis

- Sei $G = (V, \Sigma, R, S)$ eine kfG für L , dann konstruieren wir eine kfG $G' = (V', \Sigma, R', S')$, in CNF für L
- Konstruktion erfolgt in 5 Schritten durch Umbau der Regeln aus R

1. Neuer Startzustand

- Füge die Regel $S' \rightarrow S$ ein
- neuer Startzustand tritt auf keiner rechten Seite auf

Beweis

- Sei $G = (V, \Sigma, R, S)$ eine kfG für L , dann konstruieren wir eine kfG $G' = (V', \Sigma, R', S')$, in CNF für L
- Konstruktion erfolgt in 5 Schritten durch Umbau der Regeln aus R

1. Neuer Startzustand

- Füge die Regel $S' \rightarrow S$ ein
- neuer Startzustand tritt auf keiner rechten Seite auf

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

- so lange es Regeln $A \rightarrow \varepsilon$ gibt, wähle eine dieser Regeln aus
- streiche $A \rightarrow \varepsilon$
- modifiziere jede Regel mit A auf der rechten Seite so, als ob dass A auch gestrichen werden kann (alle Kombinationen!)

Beweis

- Sei $G = (V, \Sigma, R, S)$ eine kfG für L , dann konstruieren wir eine kfG $G' = (V', \Sigma, R', S')$, in CNF für L
- Konstruktion erfolgt in 5 Schritten durch Umbau der Regeln aus R

1. Neuer Startzustand

- Füge die Regel $S' \rightarrow S$ ein
- neuer Startzustand tritt auf keiner rechten Seite auf

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

- so lange es Regeln $A \rightarrow \varepsilon$ gibt, wähle eine dieser Regeln aus
- streiche $A \rightarrow \varepsilon$
- modifiziere jede Regel mit A auf der rechten Seite so, als ob dass A auch gestrichen werden kann (alle Kombinationen!)

Bsp. $X \rightarrow aABA$

wird zu $X \rightarrow aABA \mid aBA \mid aAB \mid aB$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

- Sonderfall: Regel $X \rightarrow A$
 - wird zu $X \rightarrow A \mid \varepsilon$ wenn Regel $X \rightarrow \varepsilon$ noch nicht eliminiert wurde
 - ansonsten bleibt $X \rightarrow A$ unverändert

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

- Sonderfall: Regel $X \rightarrow A$
 - wird zu $X \rightarrow A \mid \varepsilon$ wenn Regel $X \rightarrow \varepsilon$ noch nicht eliminiert wurde
 - ansonsten bleibt $X \rightarrow A$ unverändert

3. Eliminiere $A \rightarrow B$ Regeln

- streiche $A \rightarrow B$
- für alle Regeln $B \rightarrow w$ mit $w \in (V \cup \Sigma)^*$ füge die Regel $A \rightarrow w$ hinzu
- Sonderfall 1: $B \rightarrow A$
 - $A \rightarrow A$ Regel nicht aufnehmen
- Sonderfall 2: $B \rightarrow X$ und $A \rightarrow X$ wurde bereits gelöscht
 - $A \rightarrow X$ Regel nicht aufnehmen

4. Kürze zu lange rechte Seiten

- Regel $A \rightarrow x_1x_2 \dots x_k$, mit $x_i \in V \cup \Sigma$
- streiche $A \rightarrow x_1x_2 \dots x_k$
- füge folgende Regeln hinzu:

$$\begin{array}{lcl} A & \rightarrow & x_1A_1 \\ A_1 & \rightarrow & x_2A_2 \\ & \vdots & \\ A_{k-1} & \rightarrow & x_k \end{array} \quad \left. \vphantom{\begin{array}{lcl} A & \rightarrow & x_1A_1 \\ A_1 & \rightarrow & x_2A_2 \\ & \vdots & \\ A_{k-1} & \rightarrow & x_k \end{array}} \right\} A_i \text{ neue Variablen}$$

4. Kürze zu lange rechte Seiten

- Regel $A \rightarrow x_1x_2 \dots x_k$, mit $x_i \in V \cup \Sigma$
- streiche $A \rightarrow x_1x_2 \dots x_k$
- füge folgende Regeln hinzu:

$$\begin{array}{l}
 A \rightarrow x_1A_1 \\
 A_1 \rightarrow x_2A_2 \\
 \vdots \\
 A_{k-1} \rightarrow x_k
 \end{array}
 \left. \vphantom{\begin{array}{l} A \\ A_1 \\ \vdots \\ A_{k-1} \end{array}} \right\} A_i \text{ neue Variablen}$$

5. Ersetzte Regeln der Form $A \rightarrow xB$, $A \rightarrow Bx$ und $A \rightarrow xy$

- für jedes Terminal $x \in \Sigma$ füge folgende Regel ein:

$$V_x \rightarrow x \quad V_x \text{ neue Variable}$$

- ersetze $x \in \Sigma$ in den rechten Seiten durch V_x
- Bsp.: $A \rightarrow xB$ wird zu $A \rightarrow V_xB$ und $V_x \rightarrow x$

4. Kürze zu lange rechte Seiten

- Regel $A \rightarrow x_1x_2 \dots x_k$, mit $x_i \in V \cup \Sigma$
- streiche $A \rightarrow x_1x_2 \dots x_k$
- füge folgende Regeln hinzu:

$$\begin{array}{l}
 A \rightarrow x_1A_1 \\
 A_1 \rightarrow x_2A_2 \\
 \vdots \\
 A_{k-1} \rightarrow x_k
 \end{array}
 \left. \vphantom{\begin{array}{l} A \\ A_1 \\ \vdots \\ A_{k-1} \end{array}} \right\} A_i \text{ neue Variablen}$$

5. Ersetzte Regeln der Form $A \rightarrow xB$, $A \rightarrow Bx$ und $A \rightarrow xy$

- für jedes Terminal $x \in \Sigma$ füge folgende Regel ein:

$$V_x \rightarrow x \quad V_x \text{ neue Variable}$$

- ersetze $x \in \Sigma$ in den rechten Seiten durch V_x
- Bsp.: $A \rightarrow xB$ wird zu $A \rightarrow V_xB$ und $V_x \rightarrow x$

Regeländerungen erhalten Sprache und bringen G in CNF!



Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$S' \rightarrow S$

$S \rightarrow aaS \mid aSSb \mid \varepsilon$

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$S' \rightarrow S$

$S \rightarrow aaS \mid aSSb \mid \varepsilon$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$S' \rightarrow S$

$S \rightarrow aaS \mid aSSb \mid \varepsilon$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

$S' \rightarrow S \mid \varepsilon$

$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$S' \rightarrow S$

$S \rightarrow aaS \mid aSSb \mid \varepsilon$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

$S' \rightarrow S \mid \varepsilon$

$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$

3. Eliminiere $A \rightarrow B$ Regeln

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$$S' \rightarrow S$$

$$S \rightarrow aaS \mid aSSb \mid \varepsilon$$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

3. Eliminiere $A \rightarrow B$ Regeln

$$S' \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$$S' \rightarrow S$$

$$S \rightarrow aaS \mid aSSb \mid \varepsilon$$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

3. Eliminiere $A \rightarrow B$ Regeln

$$S' \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

4. Kürze zu lange rechte Seiten

Bsp. $S \rightarrow aaS \mid aSSb \mid \varepsilon$

1. Neuer Startzustand

$$S' \rightarrow S$$

$$S \rightarrow aaS \mid aSSb \mid \varepsilon$$

2. Eliminiere $A \rightarrow \varepsilon$ Regeln

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

3. Eliminiere $A \rightarrow B$ Regeln

$$S' \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab \mid \varepsilon$$

$$S \rightarrow aaS \mid aSSb \mid aa \mid aSb \mid ab$$

4. Kürze zu lange rechte Seiten

$$S' \rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \mid \varepsilon$$

$$S \rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab$$

$$X_1 \rightarrow aS$$

$$X_2 \rightarrow SX_3$$

$$X_3 \rightarrow Sb$$

$$\begin{aligned} S' &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \mid \varepsilon \\ S &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \\ X_1 &\rightarrow aS \\ X_2 &\rightarrow SX_3 \\ X_3 &\rightarrow Sb \end{aligned}$$

$$\begin{aligned} S' &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \mid \varepsilon \\ S &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \\ X_1 &\rightarrow aS \\ X_2 &\rightarrow SX_3 \\ X_3 &\rightarrow Sb \end{aligned}$$

5. Ersetzte Regeln der Form $A \rightarrow xB$, $A \rightarrow Bx$ und $A \rightarrow xy$

$$\begin{aligned}
 S' &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \mid \varepsilon \\
 S &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \\
 X_1 &\rightarrow aS \\
 X_2 &\rightarrow SX_3 \\
 X_3 &\rightarrow Sb
 \end{aligned}$$

5. Ersetzte Regeln der Form $A \rightarrow xB$, $A \rightarrow Bx$ und $A \rightarrow xy$

$$\begin{aligned}
 S' &\rightarrow V_a X_1 \mid V_a X_2 \mid V_a V_a \mid V_a X_3 \mid V_a V_b \mid \varepsilon \\
 S &\rightarrow V_a X_1 \mid V_a X_2 \mid V_a V_a \mid V_a X_3 \mid V_a V_b \\
 X_1 &\rightarrow V_a S \\
 X_2 &\rightarrow S X_3 \\
 X_3 &\rightarrow S V_b \\
 V_a &\rightarrow a \\
 V_b &\rightarrow b
 \end{aligned}$$

$$\begin{aligned}
 S' &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \mid \varepsilon \\
 S &\rightarrow aX_1 \mid aX_2 \mid aa \mid aX_3 \mid ab \\
 X_1 &\rightarrow aS \\
 X_2 &\rightarrow SX_3 \\
 X_3 &\rightarrow Sb
 \end{aligned}$$

5. Ersetzte Regeln der Form $A \rightarrow xB$, $A \rightarrow Bx$ und $A \rightarrow xy$

$$\begin{aligned}
 S' &\rightarrow V_a X_1 \mid V_a X_2 \mid V_a V_a \mid V_a X_3 \mid V_a V_b \mid \varepsilon \\
 S &\rightarrow V_a X_1 \mid V_a X_2 \mid V_a V_a \mid V_a X_3 \mid V_a V_b \\
 X_1 &\rightarrow V_a S \\
 X_2 &\rightarrow S X_3 \\
 X_3 &\rightarrow S V_b \\
 V_a &\rightarrow a \\
 V_b &\rightarrow b
 \end{aligned}$$

Chomsky Normalform

$$S \rightarrow aaS \mid aSSb \mid \varepsilon$$

Das Wortproblem für kf. Sprachen¹³

Das Wortproblem für kf. Sprachen¹³

Wortproblem

Eingabe: $w \in \Sigma^*$

Frage: $w \in L?$ für $L \in \text{CFL}$

Das Wortproblem für kf. Sprachen¹³

Wortproblem

Eingabe: $w \in \Sigma^*$

Frage: $w \in L?$ für $L \in \text{CFL}$

- Wortproblem für reguläre Sprachen trivial (Benutze DEA)

Das Wortproblem für kf. Sprachen¹³

Wortproblem

Eingabe: $w \in \Sigma^*$

Frage: $w \in L?$ für $L \in \text{CFL}$

- Wortproblem für reguläre Sprachen trivial (Benutze DEA)
- für kf. Sprachen: Algorithmus von Cocke, Younger, Kasami (**CYK-Algorithmus**)

Das Wortproblem für kf. Sprachen¹³

Wortproblem

Eingabe: $w \in \Sigma^*$

Frage: $w \in L?$ für $L \in \text{CFL}$

- Wortproblem für reguläre Sprachen trivial (Benutze DEA)
- für kf. Sprachen: Algorithmus von Cocke, Younger, Kasami (**CYK-Algorithmus**)
- Ausgangspunkt: kf. Grammatik für L in CNF

Das Wortproblem für kf. Sprachen¹³

Wortproblem

Eingabe: $w \in \Sigma^*$

Frage: $w \in L?$ für $L \in \text{CFL}$

- Wortproblem für reguläre Sprachen trivial (Benutze DEA)
- für kf. Sprachen: Algorithmus von Cocke, Younger, Kasami (**CYK-Algorithmus**)
- Ausgangspunkt: kf. Grammatik für L in CNF
- Algorithmus folgt dem Paradigma des *Dynamischen Programmierens*

CYK Algorithmus

14

8. Vorlesung

CYK Algorithmus

14

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

CYK Algorithmus

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

CYK Algorithmus

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$

CYK Algorithmus

14

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$
- **Teilprobleme des DP:** Bestimme die Mengen V_{ij}

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$
- **Teilprobleme des DP:** Bestimme die Mengen V_{ij}
- $w \in L \iff S \in V_{1,n}$, mit $|w| = n$

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$
- **Teilprobleme des DP:** Bestimme die Mengen V_{ij}
- $w \in L \iff S \in V_{1,n}$, mit $|w| = n$
- Initialisierung: $V_{ii} = \{X \mid X \rightarrow w[i, i]\}$ für $i = 1, \dots, n$

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$
- **Teilprobleme des DP:** Bestimme die Mengen V_{ij}
- $w \in L \iff S \in V_{1,n}$, mit $|w| = n$
- Initialisierung: $V_{ii} = \{X \mid X \rightarrow w[i, i]\}$ für $i = 1, \dots, n$
- Rekursiver Schritt: Bestimme V_{ij} mit Hilfe bereits bekannter Teilprobleme

- für ein Wort w bezeichnet $w[i, j]$ das Teilwort vom i ten bis j ten Zeichen

Bsp.: $w = \text{autobahn} \rightarrow w[2, 5] = \text{utob}$

- $V_{ij} := \{U \in V \mid U \Rightarrow^* w[i, j]\}$
- **Teilprobleme des DP:** Bestimme die Mengen V_{ij}
- $w \in L \iff S \in V_{1,n}$, mit $|w| = n$
- Initialisierung: $V_{ii} = \{X \mid X \rightarrow w[i, i]\}$ für $i = 1, \dots, n$
- Rekursiver Schritt: Bestimme V_{ij} mit Hilfe bereits bekannter Teilprobleme
- Abstand zwischen i und j wird in jeder *Runde* um 1 größer

Finden einer Rekursion

15

8. Vorlesung

Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit
 $U \Rightarrow XY \Rightarrow \dots \Rightarrow z$

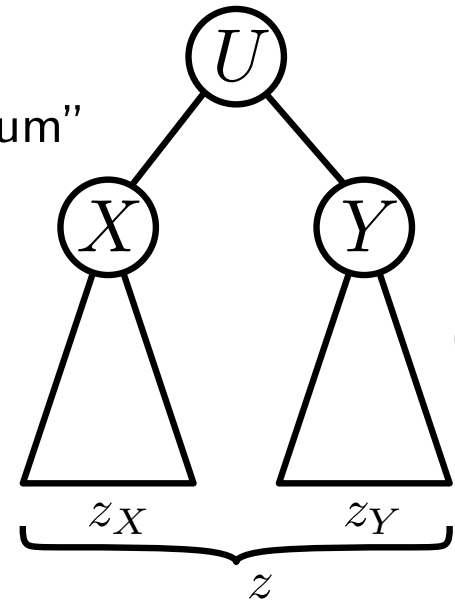
Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit

$$U \Rightarrow XY \Rightarrow \dots \Rightarrow z$$

2 Variablen da CNF

"Ableitungsbaum"



Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit

$$U \Rightarrow XY \Rightarrow \dots \Rightarrow z$$

2 Variablen da CNF

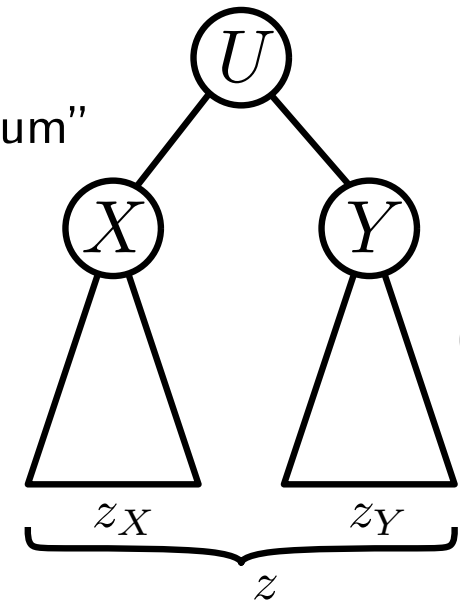
- es muss möglich sein $z = z_1 z_2$ zu zerteilen mit

1. $U \Rightarrow U_1 U_2$

2. $U_1 \Rightarrow^* z_1$

3. $U_2 \Rightarrow^* z_2$

"Ableitungsbaum"



Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit

$$U \Rightarrow XY \Rightarrow \dots \Rightarrow z$$

"Ableitungsbaum"

2 Variablen da CNF

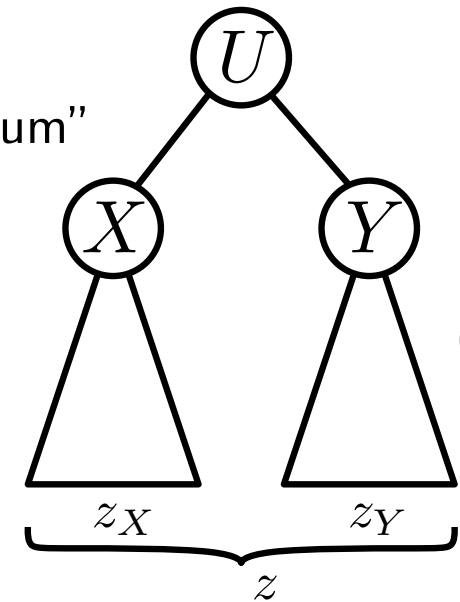
- es muss möglich sein $z = z_1 z_2$ zu zerteilen mit

1. $U \Rightarrow U_1 U_2$

2. $U_1 \Rightarrow^* z_1$

3. $U_2 \Rightarrow^* z_2$

} Teilprobleme schon gelöst



Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit

$$U \Rightarrow XY \Rightarrow \dots \Rightarrow z$$

"Ableitungsbaum"

2 Variablen da CNF

- es muss möglich sein $z = z_1 z_2$ zu zerteilen mit

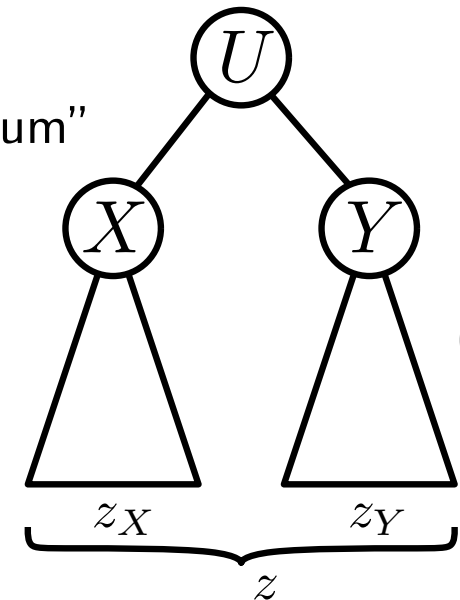
1. $U \Rightarrow U_1 U_2$

2. $U_1 \Rightarrow^* z_1$

3. $U_2 \Rightarrow^* z_2$

} Teilprobleme schon gelöst

- ich darf keine Zerteilung ausschließen \rightarrow ich muss alle Möglichkeiten z zu teilen in Betracht ziehen



Finden einer Rekursion

- sei $w[i, j] = z$, wir suchen Variablen U mit

$$U \Rightarrow XY \Rightarrow \dots \Rightarrow z$$

"Ableitungsbaum"

2 Variablen da CNF

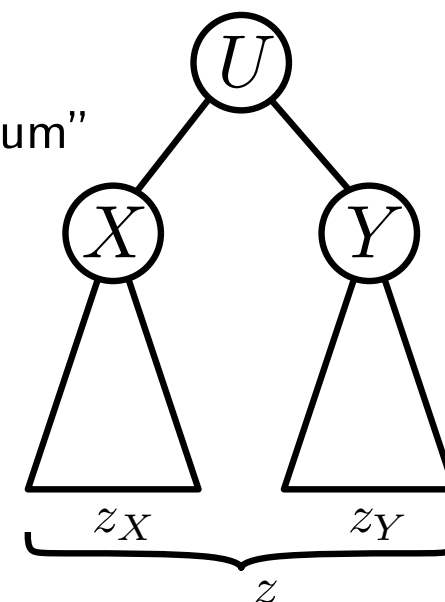
- es muss möglich sein $z = z_1 z_2$ zu zerteilen mit

1. $U \Rightarrow U_1 U_2$

2. $U_1 \Rightarrow^* z_1$

3. $U_2 \Rightarrow^* z_2$

} Teilprobleme schon gelöst



- ich darf keine Zerteilung ausschließen \rightarrow ich muss alle Möglichkeiten z zu teilen in Betracht ziehen

- **Rekursion:**

$$V_{ij} = \bigcup_{k=i}^{j-1} \{X \mid X \rightarrow AB \text{ mit } A \in V_{i,k}, B \in V_{k+1,j}\}$$

Algorithmus für $w \neq \varepsilon$

16

Algorithmus für $w \neq \varepsilon$

Datenstruktur: T , 2d Array, $T[i, j]$ enthält V_{ij}

Datenstruktur: T , 2d Array, $T[i, j]$ enthält V_{ij}

Algorithm 2: CYK Algorithmus

```
1 for  $i = 1$  to  $n$  do  $T[i, i] = \{X \in V \mid X \rightarrow w[i, i]\}$  ;
2 for  $d = 1$  to  $n - 1$  do
3   for  $i = 1$  to  $n - d$  do
4      $j = i + d$ ;
5     for  $t = i$  to  $j - 1$  do
6       for all  $A \rightarrow XY \in R$  do
7         if  $X \in T[i, t]$  and  $Y \in T[t + 1, j]$  then
8            $T[i, j] = T[i, j] \cup \{A\}$ ;
9         end
10      end
11    end
12  if  $S \in T[1, n]$  then return true;
13 else return false;
```

Datenstruktur: T , 2d Array, $T[i, j]$ enthält V_{ij}

Algorithm 2: CYK Algorithmus

```
1 for  $i = 1$  to  $n$  do  $T[i, i] = \{X \in V \mid X \rightarrow w[i, i]\}$  ; Initialisierung
2 for  $d = 1$  to  $n - 1$  do  $d$  ist Abstand  $i/j$ 
3   for  $i = 1$  to  $n - d$  do aktueller Eintrag  $T[i, j]$ 
4      $j = i + d$ ;
5     for  $t = i$  to  $j - 1$  do  $t$  ist Trennstelle
6       for all  $A \rightarrow XY \in R$  do
7         if  $X \in T[i, t]$  and  $Y \in T[t + 1, j]$  then
8            $T[i, j] = T[i, j] \cup \{A\}$ ;
9         end
10      end
11 end
12 if  $S \in T[1, n]$  then return true;
13 else return false;
```

Bsp. 1

$$\begin{aligned} S &\rightarrow AB \mid CA \\ A &\rightarrow AA \mid CB \mid a \\ B &\rightarrow AC \mid b \\ C &\rightarrow c \end{aligned}$$

$$w = cbaac$$

17

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					
4					
3					
2					
1					

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	
3			A		
2		B			
1	C				

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					<i>C</i>
4				<i>A</i>	
3			<i>A</i>		
2		<i>B</i>			
1	<i>C</i>	<i>A</i>			

Orange arrows indicate transitions: from (1,1) to (1,2) and from (2,2) to (1,2).

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	
3			A		
2		B	-		
1	C	A			

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	
2		B	-		
1	C	A			

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	
2		B	-		
1	C	A			

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	
2		B	-		
1	C	A	A		

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	
1	C	A	A		

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

17

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	
1	C	A	A		

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	
1	C	A	A		

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	
1	C	A	A	A	

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	-
1	C	A	A	A	

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	-
1	C	A	A	A	-

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	-
1	C	A	A	A	S

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					<i>C</i>
4				<i>A</i>	<i>B</i>
3			<i>A</i>	<i>A</i>	<i>S, B</i>
2		<i>B</i>	-	-	-
1	<i>C</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>S</i>

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					<i>C</i>
4				<i>A</i>	<i>B</i>
3			<i>A</i>	<i>A</i>	<i>S, B</i>
2		<i>B</i>	-	-	-
1	<i>C</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>S, B</i>

Bsp. 1

$S \rightarrow AB \mid CA$ $w = cbaac$
 $A \rightarrow AA \mid CB \mid a$
 $B \rightarrow AC \mid b$
 $C \rightarrow c$

	1	2	3	4	5
5					C
4				A	B
3			A	A	S, B
2		B	-	-	-
1	C	A	A	A	<u>S, B</u>

$S \in T[1, n]$ deshalb $w \in L(G)$