Grundlagen der theoretischen Informatik

Kurt Sieber

Fakultät IV, Department ETI
Universität Siegen

SS 2013

Vorlesung vom 23.05.2013

Kontextfreie Sprachen

Am Beispiel sieht man, dass in einer erfolgreichen Berechnung des Kellerautomaten das aktuelle Eingabewort stets aus dem aktuellen Kellerwort ableitbar ist. Diese Beobachtung bringen wir durch folgende Äquivalenz zum Ausdruck: Für alle $\alpha \in \Gamma^*$ und $w \in \Sigma^*$ gilt

$$(f, w, \alpha) \vdash_M^* (f, \varepsilon, \varepsilon) \;\Leftrightarrow\; \alpha \overset{*}{\Rightarrow}_G w \qquad (*)$$

Aus (*) folgt $L(M) = L(G)$, denn:

$$\begin{aligned}
w \in L(M) \quad &\Leftrightarrow \quad (s, w, \varepsilon) \vdash_M^* (f, \varepsilon, \varepsilon) \\
&\Leftrightarrow \quad (f, w, S) \vdash_M^* (f, \varepsilon, \varepsilon) \\
& \quad \text{weil als erster Übergangssschritt nur} \\
& \quad (s, w, \varepsilon) \vdash_M (f, w, S) \text{ in Frage kommt} \\
&\Leftrightarrow \quad S \overset{*}{\Rightarrow}_G w \\
& \quad \text{wegen } (*) \text{ mit } \alpha = S \\
&\Leftrightarrow \quad w \in L(G)
\end{aligned}$$

Es bleibt also (*) zu beweisen

$$(f, w, \alpha) \vdash_M^* (f, \varepsilon, \varepsilon) \;\Leftrightarrow\; \alpha \stackrel{*}{\Rightarrow}_G w \tag{*}$$

'$\Rightarrow$': Induktion über die Anzahl n der Übergangsschritte in $\vdash_M^*$

$n = 0$, d.h. $w = \alpha = \varepsilon$:

Dann gilt $\alpha \stackrel{*}{\Rightarrow}_G w$.

$n > 0$, d.h. $(f, w, \alpha) \vdash_M (f, v, \beta) \vdash_M^{n-1} (f, \varepsilon, \varepsilon)$ mit $v \in \Sigma^*$ und $\beta \in \Gamma^*$

Dann gilt nach Induktionsannahme $\beta \stackrel{*}{\Rightarrow}_G v$.

Wenn der erste Übergangsschritt mit (3) erfolgt, so existiert ein $a \in \Sigma$ mit $w = av$ und $\alpha = a\beta$, also gilt $\alpha = a\beta \stackrel{*}{\Rightarrow}_G av = w$.

Wenn der erste Übergangsschritt mit (2) erfolgt, so gilt $w = v$ und β entsteht aus α durch Anwendung einer Produktion $(A \to \gamma) \in P$. Dann gilt $\alpha \Rightarrow_G \beta \stackrel{*}{\Rightarrow}_G v = w$.

$$(f, w, \alpha) \vdash_M^* (f, \varepsilon, \varepsilon) \;\Leftrightarrow\; \alpha \overset{*}{\Rightarrow}_G w \qquad (*)$$

'$\Leftarrow$': Induktion über die Länge n einer *Links*ableitung $\alpha \overset{n}{\Rightarrow}_G w$

$n = 0$, d.h. $\alpha = w \in \Sigma^*$:

Dann gilt $(f, w, \alpha) = (f, w, w) \vdash_M^{|w|} (f, \varepsilon, \varepsilon)$ mit (3).

$n > 0$, d.h. es existiert ein $\alpha_1 \in \Gamma^*$ mit $\alpha \Rightarrow_G \alpha_1 \overset{n-1}{\Rightarrow}_G w$

Sei $A \to \gamma$ die Produktion im Linksableitungsschritt $\alpha \Rightarrow_G \alpha_1$, d.h. es existieren $u \in \Sigma^*, \beta \in \Gamma^*$ mit $\alpha = uA\beta$ und $\alpha_1 = u\gamma\beta$. Wegen $u \in \Sigma^*$ existiert dann ein $v \in \Sigma^*$ mit $w = uv$ und $\gamma\beta \overset{n-1}{\Rightarrow}_G v$, also

$$\begin{aligned}
(f, w, \alpha) = (f, uv, uA\beta) \quad & \vdash_M^{|u|} \quad (f, v, A\beta) \text{ mit (3)} \\
& \vdash_M \quad (f, v, \gamma\beta) \text{ mit (2)} \\
& \vdash_M^* \quad (f, \varepsilon, \varepsilon) \text{ nach Induktionsannahme } \Box
\end{aligned}$$

Die Umkehrung von Satz 2.58 gilt ebenfalls.

Satz 2.59 *Zu jedem Kellerautomaten M lässt sich eine kontextfreie Grammatik G konstruieren mit $L(G) = L(M)$.*

Beweis: s. Literatur $\square$

Also erhalten wir eine zweite Charakterisierung für die Klasse der kontextfreien Sprachen.

Satz 2.60 *Eine Sprache $L \subseteq \Sigma^*$ ist genau dann kontextfrei, wenn es einen Kellerautomaten M gibt mit $L = L(M)$.*

Beweis: Das folgt unmittelbar aus den Sätzen 2.58 und 2.59. $\square$

Abschlusseigenschaften

Satz 2.61 *Die Klasse $\mathcal{L}_{kf}$ der kontextfreien Sprachen ist abgeschlossen unter den Operationen $\cup, \circ, ^*$ und $^+$, d.h. wenn $L_1, L_2 \subseteq \Sigma^*$ kontextfrei sind, dann sind auch die Sprachen*

1. $L_1 \cup L_2$
2. $L_1 \circ L_2$
3. L_1^*
4. L_1^+

kontextfrei. Darüber hinaus gibt es Algorithmen, um Grammatiken für diese Sprachen aus den Grammatiken für L_1 und L_2 zu konstruieren.

Beweis:

Seien $G_i = (\Sigma, N_i, S_i, P_i)$ $(i = 1,2)$ kontextfreie Grammatiken mit $L(G_i) = L_i$. Wir dürfen annehmen, dass $N_1 \cap N_2 = \emptyset$.

1. siehe Übungsblatt 5, Aufgabe 5
2. Sei $G = (\Sigma, N, S, P)$, wobei
   - S ein neues Zeichen ist, d.h. $S \notin N_1 \cup N_2 \cup \Sigma$,
   - $N = N_1 \cup N_2 \cup \{S\}$,
   - $P = \{S \to S_1 S_2\} \cup P_1 \cup P_2$.

   Dann gilt $L(G) = L_1 \circ L_2$.

   '$\supseteq$': Sei $w \in L_1 \circ L_2$, d.h. $w = w_1 w_2$ mit $w_i \in L_i$ für $i = 1,2$.

   Dann gilt $S_i \overset{*}{\Rightarrow}_{G_i} w_i$ und damit auch $S_i \overset{*}{\Rightarrow}_G w_i$ für $i = 1,2$, und es folgt $S \Rightarrow_G S_1 S_2 \overset{*}{\Rightarrow}_G w_1 S_2 \overset{*}{\Rightarrow}_G w_1 w_2 = w$, d.h. $w \in L(G)$.

'$\subseteq$': Sei $w \in L(G)$, d.h. es gibt eine Linksableitung $S \stackrel{*}{\Rightarrow}_G w$.

Da S ein neues Zeichen ist, kann der erste Ableitungsschritt nur mit der neuen Produktion $S \to S_1S_2$ erfolgt sein, also hat die gesamte Ableitung die Form $S \Rightarrow S_1S_2 \stackrel{*}{\Rightarrow}_G w_1S_2 \stackrel{*}{\Rightarrow}_G w_1w_2 = w$, wobei $S_1 \stackrel{*}{\Rightarrow}_G w_1$ und $S_2 \stackrel{*}{\Rightarrow}_G w_2$.

Wegen $N_1 \cap N_2 = \emptyset$ können in $S_1 \stackrel{*}{\Rightarrow}_G w_1$ nur Ableitungsschritte für G_1 vorkommen und in $S_2 \stackrel{*}{\Rightarrow}_G w_2$ nur solche für G_2.

Also gilt $S_1 \stackrel{*}{\Rightarrow}_{G_1} w_1$ und $S_2 \stackrel{*}{\Rightarrow}_{G_2} w_2$, d.h. $w_1 \in L_1, w_2 \in L_2$ und damit $w = w_1w_2 \in L_1 \circ L_2$.

3. Sei $G = (\Sigma, N, S, P)$, wobei

   $S \notin N_1 \cup \Sigma$,

   $N = N_1 \cup \{S\}$,

   $P = \{S \to S_1S,\ S \to \varepsilon\} \cup P_1$.

Dann gilt $L(G) = L_1^*$.

'$\supseteq$': Sei $w \in L_1^*$, d.h. es existieren $n \geq 0$ und $w_1, \ldots, w_n \in L_1$ mit $w = w_1 \ldots w_n$.

Dann gilt $S \Rightarrow S_1 S \Rightarrow \ldots \Rightarrow S_1^n S \Rightarrow S_1^n$, und wegen $S_1 \overset{*}{\Rightarrow}_G w_i$ für $i = 1, \ldots n$ folgt $S \overset{*}{\Rightarrow}_G w_1 \ldots w_n = w$, also $w \in L(G)$.

'$\subseteq$': Sei $w \in L(G)$. Durch Induktion über die Länge einer Linksableitung $S \overset{*}{\Rightarrow}_G w$ beweisen wir $w \in L_1^*$.

Da S ein neues Zeichen ist, kann der erste Ableitungsschritt von $S \overset{*}{\Rightarrow}_G w$ nur mit einer der neuen Produktionen $S \to \varepsilon$ oder $S \to S_1 S$ erfolgt sein. Im ersten Fall ist $w = \varepsilon \in L_1^*$. Im zweiten Fall ist die gesamte Ableitung von der Form $S \Rightarrow S_1 S \overset{*}{\Rightarrow}_G w_1 S \overset{*}{\Rightarrow}_G w_1 w' = w$, wobei $S_1 \overset{*}{\Rightarrow}_G w_1$ und $S \overset{*}{\Rightarrow}_G w'$.

Da $S_1 \overset{*}{\Rightarrow}_G w_1$ nur Ableitungsschritte für G_1 enthalten kann, gilt $w_1 \in L(G_1) = L_1$, und nach Induktionssannahme gilt $w' \in L_1^*$.

Also folgt $w = w_1 w' \in L_1 \circ L_1^* \subseteq L_1^*$.

4. Die Abgeschlossenheit unter $^+$ folgt aus der Abgeschlossenheit unter $\circ$ und *, weil $L_1^+ = L_1 \circ L_1^*$. $\square$

Es ist kein Zufall, dass Durchschnitt und Komplement in Satz 2.61 fehlen. Wir werden später sehen, dass der Durchschnitt zweier kontextfreier Sprachen im allgemeinen *nicht* kontextfrei ist.

Daraus folgt sofort, dass auch das Komplement einer kontextfreien Sprache im allgemeinen *nicht* kontextfrei ist, denn aus der Abgeschlossenheit unter Vereinigung und Komplement würde sich die Abgeschlossenheit unter Durchschnitt ergeben.

Für den Durchschnitt gilt eine *schwächere* Aussage, die sich besser mit Automaten (als mit Grammatiken) beweisen lässt:

Satz 2.62 *Wenn $L_1 \subseteq \Sigma^*$ kontextfrei und $L_2 \subseteq \Sigma^*$ regulär ist, dann ist $L_1 \cap L_2$ kontextfrei.*

Beweis:

Sei $M_1 = (\Sigma, \Gamma, Q_1, s_1, F_1, \Delta_1)$ ein Kellerautomat mit $L(M_1) = L_1$ und sei $A_2 = (\Sigma, Q_2, s_2, F_2, \delta)$ ein DEA mit $L(A_2) = L_2$.

Wir konstruieren einen Kellerautomaten M mit $L(M) = L_1 \cap L_2$.

Idee: M arbeitet wie M_1 und simuliert "parallel dazu" noch die Übergänge des endlichen Automaten A_2.

Diese "Parallelverarbeitung" bringt man dadurch zum Ausdruck, dass man auf der Zustandsmenge $Q_1 \times Q_2$ arbeitet: In der ersten Komponente merkt man sich den aktuellen Zustand von M_1, in der zweiten Komponente den von A_2.

Sei also $M = (\Sigma, \Gamma, Q_1 \times Q_2, (s_1, s_2), F_1 \times F_2, \Delta)$ mit

$$\Delta = \{(((p_1, p_2), u, \beta), ((q_1, q_2), \gamma)) \mid ((p_1, u, \beta), (q_1, \gamma)) \in \Delta_1 \text{ und } (p_2, u) \vdash^*_{A_2} (q_2, \varepsilon)\}$$

Δ ist gerade so konstruiert, dass in jedem Übergangsschritt von M ein Übergangsschritt von M_1 und eine entsprechende Folge von Übergangsschritten von A_2 simuliert werden, d.h. es gilt stets:

$$((p_1, p_2), u, \alpha) \vdash_M ((q_1, q_2), \varepsilon, \alpha') \;\Leftrightarrow\; (p_1, u, \alpha) \vdash_{M_1} (q_1, \varepsilon, \alpha') \text{ und } (p_2, u) \vdash^*_{A_2} (q_2, \varepsilon)$$

Das ergibt sich unmittelbar aus der Definition von Δ und der Definition der möglichen Übergangsschritte eines Kellerautomaten.

Kontextfreie Sprachen

Der gleiche Zusammenhang gilt dann auch für *Folgen* von Übergangs-schritten:

$$((p_1,p_2),u,\alpha) \vdash_M^* ((q_1,q_2),\varepsilon,\alpha') \;\Leftrightarrow\; (p_1,u,\alpha) \vdash_{M_1}^* (q_1,\varepsilon,\alpha') \text{ und } (p_2,u) \vdash_{A_2}^* (q_2,\varepsilon)$$

Das ergibt sich leicht durch Induktion über die Länge der Folge.

Daraus erhält man schließlich das gewünschte Ergebnis:

$$\begin{aligned}
w \in L(M) \;&\Leftrightarrow\; \text{es existiert ein } (q_1,q_2) \in F \text{ mit} \\
&\qquad ((s_1,s_2),w,\varepsilon) \vdash_M^* ((q_1,q_2),\varepsilon,\varepsilon) \\
&\Leftrightarrow\; \text{es existieren } q_1 \in F_1, q_2 \in F_2 \text{ mit} \\
&\qquad (s_1,w,\varepsilon) \vdash_{M_1}^* (q_1,\varepsilon,\varepsilon) \\
&\qquad \text{und } (s_2,w) \vdash_{A_2}^* (q_2,\varepsilon) \\
&\Leftrightarrow\; w \in L(M_1) \cap L(A_2) = L_1 \cap L_2
\end{aligned}$$

□

Entscheidbarkeitsfragen

Wir haben gesehen: Viele Fragestellungen über reguläre Sprachen sind entscheidbar, d.h. es gibt (einfache, manchmal auch effiziente) Algorithmen, die stets die richtige Antwort liefern.

Gilt das auch für kontextfreie Sprachen?

Wichtigste Fragestellung ist das *Wortproblem*:

1. Das *spezielle* Wortproblem für *eine* kontextfreie Grammatik G:

   Eingabe: Ein Wort $w \in \Sigma^*$.

   Frage: Ist $w \in L(G)$?

2. Das *allgemeine* Wortproblem für kontextfreie Grammatiken:

   Eingabe: Eine kontextfreie Grammatik G und ein Wort $w \in \Sigma^*$.

   Frage: Ist $w \in L(G)$?

Kontextfreie Sprachen

Für die Praxis ist das spezielle Wortproblem wichtig, denn das ist die Frage, die ein Parser für die Sprache $L(G)$ beantworten muss: Ist die Zeichenreihe, die der Programmierer eingibt, ein syntaktisch korrektes Programm? Diese Frage muss nicht nur entscheidbar sein, sondern sie muss sich *effizient* lösen lassen.

Wir zeigen hier, dass sogar das allgemeine Wortproblem entscheidbar ist (allerdings nicht sehr effizient).

Lösungsansatz:

Um $w \in L(G)$ zu überprüfen, sucht man systematisch nach einer Ableitung für w aus dem Startsymbol S. Man kann z.B. erst alle Wörter $u \in (N \cup \Sigma)^*$ bestimmen, die in einem Schritt aus S ableitbar sind, dann die, die in zwei Schritten ableitbar sind usw.

Wenn w dabei irgendwann auftaucht, ist die Antwort "ja". Aber wann kann man die Antwort "nein" geben, d.h. wann kann man sicher sein, dass w nicht mehr auftaucht?

Wenn wir wüssten, dass bei jedem Ableitungsschritt mindestens ein Zeichen hinzukommt, dann hätten wir ein Abbruchkriterium: Dann bräuchten wir nur Ableitungen zu untersuchen, die höchstens die Länge $|w|$ haben.

Probleme bereiten also Ableitungsschritte, bei denen das Wort *nicht* länger wird. Solche Ableitungsschritte entstehen, wenn man Produktionen $A \to \gamma$ mit $|\gamma| \leq 1$ anwendet.

Definition 2.63

- *Eine Produktion der Form $A \to \varepsilon$ (mit $A \in N$) heißt* ε-Produktion.
- *Eine Produktion der Form $A \to B$ (mit $A, B \in N$) heißt* Einheitsproduktion.

Wir werden zeigen, dass man solche Produktionen (bis auf eine) aus einer kontextfreien Grammatik entfernen kann, ohne dass sich die von der Grammatik erzeugte Sprache verändert.

Satz 2.64 *Zu jeder kontextfreien Grammatik* $G = (\Sigma, N, S, P)$ *kann man eine äquivalente kontextfreie Grammatik* $G' = (\Sigma, N, S, P')$ *mit folgenden Eigenschaften konstruieren:*

1. P' *enthält* höchstens eine ε*-Produktion, nämlich* $S \to \varepsilon$*,*
2. *diese* ε*-Produktion wird* nur *zur Ableitung von* ε *benötigt.*

Beweis:

Die Konstruktion von G' lässt sich in zwei Phasen unterteilen.

In der *ersten Phase* wird P schrittweise zu einer neuen Produktionenmenge P'' *erweitert*, wobei jeder einzelne Erweiterungsschritt so aussieht:

Wenn bereits zwei Produktionen der Form $B \to \beta A \gamma$ und $A \to \varepsilon$ vorhanden sind (d.h. wenn sie in P liegen oder durch vorhergehende Erweiterungsschritte hinzugekommen sind), dann wird die Produktion $B \to \beta\gamma$ aufgenommen (die sogar eine ε-Produktion sein kann).

Diese Erweiterungsschritte führt man so lange durch, bis keine neuen Produktionen mehr entstehen.

Das Verfahren *terminiert*, weil die rechte Seite einer neuen Produktion stets kürzer ist als die rechte Seite einer der Produktionen in P. Damit kommen von vornherein nur endlich viele unterschiedliche rechte Seiten und damit auch nur endlich viele unterschiedliche neue Produktionen in Frage.

Das Verfahren ist auch *korrekt*, d.h. durch die neu hinzugenommenen Produktionen wird die erzeugte Sprache nicht größer:

Jede Anwendung einer neuen Produktion $B \to \beta\gamma$ entspricht nämlich einer Anwendung von $B \to \beta A\gamma$, gefolgt von einer Anwendung von $A \to \varepsilon$, d.h. man kann jede Anwendung einer Produktion in P'' letztendlich durch mehrere (aufeinanderfolgende) Anwendungen von Produktionen der ursprünglichen Menge P ersetzen.

In der *zweiten Phase* wird P'' zu P' *verkleinert*, indem man einfach alle ε-Produktionen außer $S \to \varepsilon$ aus P'' entfernt.

Es bleibt zu zeigen, dass die erzeugte Sprache durch das Entfernen dieser Produktionen nicht kleiner wird.

Dazu geben wir an, wie man eine Ableitung $S \stackrel{*}{\Rightarrow} w$ eines Wortes $w \in \Sigma^*$ in P'' schrittweise zu einer Ableitung in P' umformen kann.

Jeder einzelne Umformungsschritt sieht so aus:

Wenn im ersten Ableitungsschritt von $S \stackrel{*}{\Rightarrow} w$ eine ε-Produktion angewandt wird, so ist $w = \varepsilon$ und die Ableitung liegt bereits in P'.

Wenn in einem späteren Schritt von $S \stackrel{*}{\Rightarrow} w$ eine ε-Produktion $A \to \varepsilon$ angewandt wird, dann ist das Nichtterminalzeichen A irgendwann vorher durch eine Produktion $B \to \beta A \gamma$ entstanden.

Kontextfreie Sprachen

Also kann man sich die Anwendung von $A \to \varepsilon$ ersparen, indem man in diesem früheren Schritt anstelle von $B \to \beta A \gamma$ gleich die Produktion $B \to \beta\gamma$ anwendet (die wir ja in P'' aufgenommen haben).

Den soeben geschilderten Umformungsschritt wiederholt man so oft wie möglich.

Das Verfahren terminiert, weil die Ableitung bei jedem Umformungsschritt um einen Ableitungsschritt kürzer wird (was ja nicht unendlich oft passieren kann).

Also erreicht man irgendwann eine Situation, in der keine Umformung mehr möglich ist.

Das bedeutet aber, dass die Ableitung entweder keine Anwendungen von ε-Produktionen mehr enthält oder nur noch aus dem Ableitungsschritt $S \Rightarrow \varepsilon$ besteht.

In beiden Fällen ist es dann eine Ableitung in P'. $\square$

Satz 2.65 *Zu jeder kontextfreien Grammatik $G = (\Sigma, N, S, P)$ kann man eine* äquivalente *kontextfreie Grammatik $G' = (\Sigma, N, S, P')$ konstruieren, die* keine *Einheitsproduktionen enthält (und falls G durch das Verfahren in Satz 2.64 entstanden ist, kann man die Konstruktion so durchführen, dass keine neuen ε-Produktionen entstehen).*

Beweis:

Die Konstruktion von G' verläuft wieder in zwei Phasen.

In der *ersten Phase* wird P schrittweise zu P'' *erweitert*, wobei jeder einzelne Erweiterungsschritt so aussieht:

Wenn bereits zwei Produktionen der Form $A \to B$ und $B \to \gamma$ vorhanden sind, dann wird die Produktion $A \to \gamma$ aufgenommen (wobei man hier *keine* ε-Produktionen aufnimmt, falls G nach dem Verfahren in Satz 2.64 entstanden ist).

Diese Erweiterungsschritte führt man so lange durch, bis keine neuen Produktionen mehr entstehen.

Das Verfahren *terminiert*, weil sich die neuen Produktionen von den alten nur durch das Nichtterminalzeichen auf der linken Seite unterscheiden, also können aus jeder Produktion in P höchstens $|N| - 1$ neue entstehen.

Das Verfahren ist *korrekt* weil auch hier jede neue Produktion nur eine 'Abkürzung' für eine Folge von alten Produktionen ist.

In der *zweiten Phase* wird P'' zu P' *verkleinert*, indem man alle Einheitsproduktionen aus P'' entfernt.

Hier muss man sich wieder davon überzeugen, dass die erzeugte Sprache nicht kleiner wird.

Dazu zeigt man, dass sich die Einheitsproduktionen aus jeder Ableitung $S \stackrel{*}{\Rightarrow} w$ eines Wortes $w \in \Sigma^*$ entfernen lassen.

Sei $A \to B$ die *letzte* Einheitsproduktion, die in der Ableitung $S \stackrel{*}{\Rightarrow} w$ angewandt wird.

Weil das entstehende Nichtterminalzeichen B nicht in w vorkommen kann, muss weiter rechts in der Ableitung eine Produktion $B \to \gamma$ auf *dieses* Zeichen B angewandt werden (dabei ist $B \to \gamma$ *keine* ε-Produktion, falls G durch das Verfahren in Satz 2.64 entstanden ist).

Also kann man diese Anwendung von $B \to \gamma$ einsparen, indem man gleich $A \to \gamma$ anstelle von $A \to B$ anwendet.

Man beachte, dass dabei eine Einheitsproduktion aus der Ableitung verschwindet: Per Definition ist $A \to B$ ja die *letzte* Einheitsproduktion in der Ableitung, also ist $B \to \gamma$ und damit auch das neu entstehende $A \to \gamma$ *keine* Einheitsproduktion.

Durch Wiederholung dieses Umformungsschrittes kann man also nach und nach alle Anwendungen von Einheitsproduktionen aus der Ableitung $S \stackrel{*}{\Rightarrow} w$ entfernen, und erhält so eine Ableitung mit der neuen Produktionenmenge P'. $\square$

Satz 2.66 *Für jede der folgenden Fragestellungen gibt es einen Entscheidungsalgorithmus.*

1. Allgemeines Wortproblem für kontextfreie Grammatiken

   Eingabe: Eine kontextfreie Grammatik G und ein Wort w.

   Frage: Ist $w \in L(G)$?

2. Leerheitsproblem für kontextfreie Grammatiken

   Eingabe: Eine kontextfreie Grammatik G.

   Frage: Ist $L(G) = \emptyset$?.

Beweis:

1. Wegen Satz 2.64 und Satz 2.65 dürfen wir annehmen, dass G keine Einheitsproduktionen enthält und höchstens eine ε-Produktion $S \to \varepsilon$, die nur zur Ableitung von ε benötigt wird.

Kontextfreie Sprachen

Im Falle $w = \varepsilon$ brauchen wir nur nachzusehen, ob $S \to \varepsilon$ in der Produktionenmenge liegt.

Im Falle $w \neq \varepsilon$ können wir die Länge einer Ableitung $S \stackrel{*}{\Rightarrow} w$ nach oben abschätzen.

In $S \stackrel{*}{\Rightarrow} w$ werden ja nur Produktionen der Form $A \to \gamma$ mit $|\gamma| > 1$ oder $A \to a$ mit $a \in \Sigma$ angewandt.

Jeder Ableitungssschritt mit einer Produktion der ersten Form macht das bereits abgeleitete Wort länger, also können höchstens $|w| - |S| = |w| - 1$ solche Ableitungsschritte in $S \stackrel{*}{\Rightarrow} w$ vorkommen.

Jeder Ableitungsschritt mit einer Produktion $A \to a$ produziert ein Terminalzeichen, das nicht mehr entfernt werden kann, also können höchstens $|w|$ solche Ableitungsschritte vorkommen.

Damit beträgt die Gesamtlänge der Ableitung $S \stackrel{*}{\Rightarrow} w$ höchstens $2|w| - 1$.