Vorlesung Baumautomaten (Mitschrift)

Benedikt Elßmann (3720358)
be57xocu@studserv.uni-leipzig.de

Universität Leipzig

19. April 2019

Inhaltsverzeichnis

0 Einleitung

Automaten lesen Wörter $w = a_1 \dots a_n$ und geben "accept" aus oder nicht.
Dafür gibt es Erweiterungen, wie etwa:

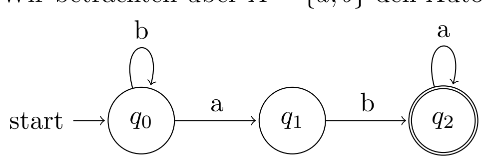
- gewichtete Automaten, das heißt der Output ist ein Semiringelement
- Automaten mit Gedächtnis (Stack)
- Automaten über anderen Strukturen
  - ω-Wörter $w = a_1 \dots a_n$
  - Graphen
  - Bäume
  - Kombinationen dieser

Typische Fragestellungen:

- Ausdrucksstärke
- Darstellung als rationale Ausdrücke (Kleene)
- Darstellung als Grammatik
- Darstellung als Logik

1 Bäume und Baumautomaten

Wir betrachten über $A = \{a, b\}$ den Automaten $\mathcal{A}$:

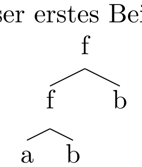


mit $L(\mathcal{A}) = b^*aba^*$.
Betrachtung des Wortes $w = baba \in L(\mathcal{A})$:

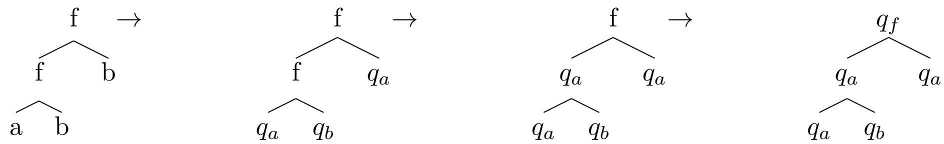
Der eindeutige erfolgreiche Lauf für w lässt sich darstellen als:

$q_0baba \to bq_0aba \to baq_1ba \to babq_2a \to babaq_2 \in F$ (Finalzustand)

Baumautomaten funktionieren analog. Unser erstes Beispiel wird



Akzeptiert mit dem Lauf:



mit $q_f \in F$

1.1 Definition Rangalphabet

Ein paar (Σ, rk), wobei Σ eine endliche Menge von Symbolen und $rk : \Sigma \to \mathbb{N}$ eine Abbildung ist, heißt Rangalphabet.
Für $f \in \Sigma$ heißt $rk(f)$ der Rang (oder die Stelligkeit) von f.

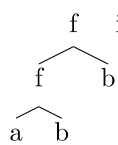
Intuitiv: $rk(f)$ ist die Anzhal der Kinder von f in einem Baum.
Insbesondere ist die Anzahl der Kinder für jedes Symbol fest.

Gilt $rk(f) = n$, schreiben wir auch $f^{(n)}$ statt f. wir schreiben:

- 0-stellige Symbole (Konstanten) $a, b, \ldots$
- unär, binär, $\ldots f, g, \ldots$

Wir setzen $\Sigma^{(n)} = \{f \in \Sigma | rk(f) = n\}$

In



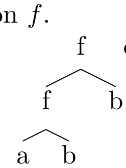
ist also $rk(f) = 2, rk(b) = 0$

1.2 Definition Term, Tree

Sei (Σ, rk) ein Rangalphabet. Die Menge T_Σ der Bäume üeber Σ ist induktiv definiert durch:

- $\Sigma^0 \subseteq T_\Sigma$
- $f^{(n)} \in \Sigma$. $t_1, \ldots, t_n \in T_\Sigma$, dann ist $f(t_1, \ldots, t_n) \in T_\Sigma$

Intuitiv sind $t_1, \ldots, t_n$ die Kinder von f.
Z.B. ist



der Term $f(f(a, b), b)$.

1.3 Definition Höhe

Sei (Σ, rk) ein Rangalphabet. Die Höhe ht ist gegeben durch:

- für $a^{(0)} \in \Sigma : ht(a) = 1$.
- für $f(t_1, \ldots, t_n) \in T_\Sigma : ht(f) = 1 + max\{ht(t_i) | i \in \{i, \ldots, n\}\}$

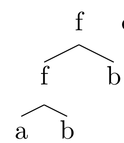
Ziel: Zugriff auf einen Knoten innterhalb eines Baumes und deren Label.
Dafür ordenen wir den Knoten Positionen zu. Das geht induktiv wie foelgt:

1.4 Definition Position

Sei (Σ, rk) ein Rangalphabet. Die Positionenmenge ist definiert durch:

- für $a^{(0)} \in T_\Sigma$ ist $Pos(a) = \{\varepsilon\}$
- für $f(t_1, \ldots, t_n) \in T_\Sigma$ ist $Pos(f(t_1, \ldots, t_n)) = \{\varepsilon\}1 \cdot Pos(t_1) \cup \cdots \cup n \cdot Pos(t_n)$

Beispiel:
Betrachtung von $f(f(a, b), b)$ bzw.



der Term $f(f(a, b), b)$

$Pos(f) = \{\varepsilon, 1, 2, 1.1, 1.2\}$

1.5 Definition der Label an den Positionen

Für einen Term der Form $t = f(t_1, \ldots, t_n)$ ist das Symbol $t(p)$ in t an p-ter Position induktiv definert durch:

- $t(\varepsilon) = f$
- $t(ip) = t_i(p), i \in \{1, \ldots, n\}$

Beispiel: Betrachtung von $f(f(a, b), b)$
Dann ist
$t(\varepsilon) = f$
$t(1) = t(1 \cdot \varepsilon) = t_1(\varepsilon) = f$
$t(2) = t(2 \cdot \varepsilon) = t_2(\varepsilon) = b$
$t(1.1) = t_1(1) = a$
$t(1.2) = t_2(1) = b$

1.6 Definition Sub-Baum

Für T_Σ ist ein Sub-Baum $t_{|p}$ an p-ter Position wie folgt definiert:

- $Pos(t_{|p}) = \{i | pi \ \in Pos(t)\}$

- $\forall q \in Pos(t_{|p}$ ist $t_{|p}(q) = t(pq)$

Wir schreiben $t[u]_p$ für den Baum, der entsteht, wenn man in t den sub-Baum $t_{|p}$ durch n ersetzt.

Beispiel: $f(f(a,b),b)$ bzw.

f
f b
a b

$t_{|1} = f(a,b)$

f
a b

$t_{|2} = t(1.2) = b$

$u = g(b,a)$

g , dann ist
b a

$t[u]_{|1} = f(g(b,a),b)$

f
g b
b a

1.7 Definition Baumautomat

Ein Buamautomat $\mathcal{A}$ ist ein 4-Tupel (Q, Σ, F, Δ), wobei:
$Q \ldots$ endliche Menge an Zusänden
$\Sigma \ldots$ Rangalphabet, wobei $\Sigma \cup Q \neq \emptyset$
$F \cdots \subseteq Q$ Finalzustände
$\Delta \ldots$ Menge von Regeln

$r : f(q_1 \ldots q_n) \to q$
für $q, q_1, \ldots, q_n \in Q$, für $a^{(0)} \in T_\Sigma : a \to q$

Beispiel:
$\mathcal{A} = \{\{q_a, q_b, q_f\}, \{a^{(0)}, b^{(0)}, f^{(2)}\}, \{q_f\}, \Delta\}$
mit $\Delta = \{a \to q_a, b \to q_b, f(q_a, q_b) \to q_a, f(q_a, q_b), f(q_a, q_b) \to q_f\}$

1.8 Definition Lauf/Run

Sei $\mathcal{A} = (Q, \Sigma, F, \Delta)$ ein Baumautomat und $t \in T_\Sigma$. Ein Lauf r für t von $\mathcal{A}$ ist ein Term mit

- $Pos(r) = Pos(t)$
- Ist $t(p) = a$ ein Blatt, dann ist $r(p) = q_a$, nur wenn $(a \to q_a) \in \Delta$
- Ist $t(p) = f^{(m)}$, dann ist $r(p) = q$, wenn $(f(q_1, \ldots, q_n) \to q) \in \Delta$ und $r(p_i) = q_i$, $i \in \{1, \ldots, n\}$

Ein Lauf ist erfolgreich, wenn $r(\varepsilon) \in F$. Der Automat $\mathcal{A}$ akzeptiert t, falls es einen erfolgreichen Lauf für t von $\mathcal{A}$ gibt.
Wir bezeichnen mit $L(\mathcal{A}) = \{t \in T_\Sigma | \mathcal{A} \text{ akzeptiert } t\}$ die von $\mathcal{A}$ erkannte Baumsprache. Eine Sprache $L \subseteq T_\Sigma$ heißt erkennbar, falls ein Baumautomat $\mathcal{A}$ existiert mit $L = L(\mathcal{A})$.

Um einzelne Schritte von Baumautomaten zu formalisieren, betrachten wir die *move relation* $\rightarrow_\mathcal{A}$, definiert wie folgt:
Gegeben sei $\mathcal{A} = (Q, \Sigma, F, \Delta)$, dann ist $t \rightarrow_\mathcal{A} t'$ mit $t, t' \in T_{\Sigma \cup Q}$, falls

- $t(p) = f^{(n)}$
- $t(pi) = q_i$ für $i \in \{1, \dots, n\}$ und p_i sind Blätter
- $(f(q_1, \dots, q_n) \rightarrow q) \in \Delta$
- und $t' = t[q]_p$

Mit $\rightarrow_\mathcal{A}^*$ bezeichnen wir die transitive Hülle von $\rightarrow_\mathcal{A}$.

1.9 Lemma

Sei $\mathcal{A} = (Q, \Sigma, F, \Delta)$ ein Baumautomat. Dann ist $L(\mathcal{A}) = \{t \in T_\Sigma | t \rightarrow_\mathcal{A}^* q \text{ mit } q \in F\} (= Z)$

Beweis: „$L(\mathcal{A}) \subseteq Z$“:

Wir zeigen: Es existiert ein Run r für t von $\mathcal{A}$ mit $r(\varepsilon) = q$, dann ist $t \rightarrow_\mathcal{A}^* q$

Inuktionsannahme:
$t = a^{(0)} \in T_\Sigma$. Dann gilt $a \in L(\mathcal{A})$, falls ein Lauf r existiert mit $r(a) = q_a$ und $(a \rightarrow q_a) \in \Delta$. Dann folgt $a \rightarrow_\mathcal{A}^* q_a$.
Sei nun $t = f(t_1, \dots, t_n)$

Induktionsvoraussetzung:
Falls für $t_1, \dots, t_n$ Läufe r_i existieren mit $r_i(\varepsilon) = q_i$, dann gilt auch $t_i \rightarrow_\mathcal{A}^* q_i$ mit $i \in \{1, \dots, n\}$

Induktionsschritt:
zu zeigen: Es existiert ein Lauf r für t mit $r(\varepsilon) = q$, dann $t \rightarrow_\mathcal{A}^* q$.
Sei also r ein Lauf mit $r(\varepsilon) = q$. Dann ist $r(i) = q_i$, $i \in \{1, \dots, n\}$, mit $(f(q_1, \dots, q_n) \rightarrow q) \in \Delta$.
Laut Induktionsvoraussetzung gilt nun, $t_i \rightarrow_\mathcal{A}^* q_i$, $i \in \{1, \dots, n\}$.
Damit $t = f(t_1, \dots, t_n) \rightarrow_\mathcal{A}^* f(q_1, t_2, \dots, t_n) \rightarrow_\mathcal{A}^* \cdots \rightarrow_\mathcal{A}^* f(q_1, \dots, q_n)$
Des weiteren haben wir die regel $f(q_1, \dots, q_n) \rightarrow q$, das heißt $f(q_1, \dots, q_n) \rightarrow_\mathcal{A}^* q$.

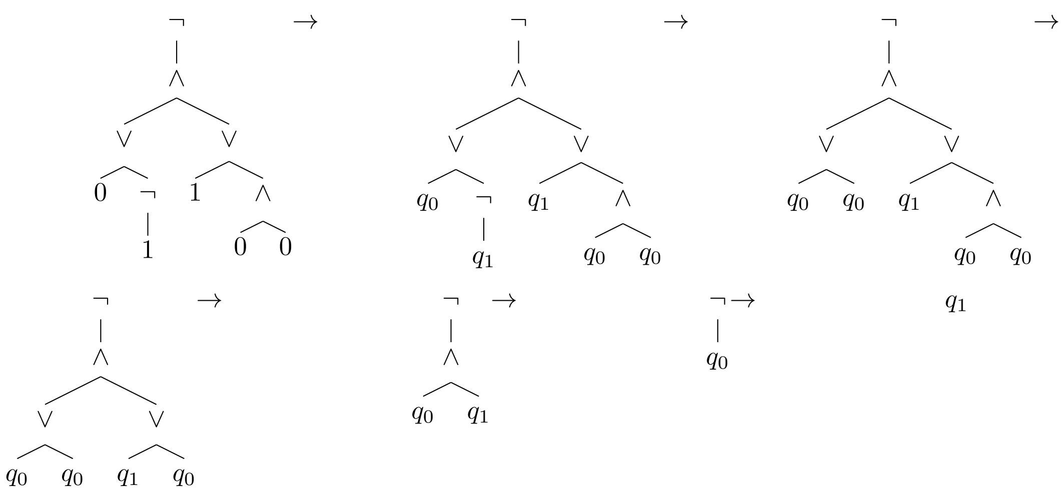
Insgesamt also $t \rightarrow_\mathcal{A}^* q$

Beweis: „$L(Z \subseteq \mathcal{A})$“: analog

Einige Beispiele für Baumautomaten:

1. Sei $B = (\{q_0, q_1\}, \{0^{(0)}, 1^{(0)}, \neg^{(1)}, \wedge^{(2)}, \vee^{(2)}\ \{q_1\}, \Delta\}$ mit
$\Delta = \{0 \to q_0, 1 \to q_1,$
$\neg(q_0) \to q_1, \neg(q_1) \to q_0,$
$\wedge(q_0, q_0) \to q_0, \wedge(q_0, q_1) \to q_0, \wedge(q_1, q_0) \to q_0, \wedge(q_1, q_1) \to q_1$
$\vee(q_0, q_0) \to q_0, \vee(q_0, q_1) \to q_1, \vee(q_1, q_0) \to q_1, \vee(q_1, q_1) \to q_1\}$

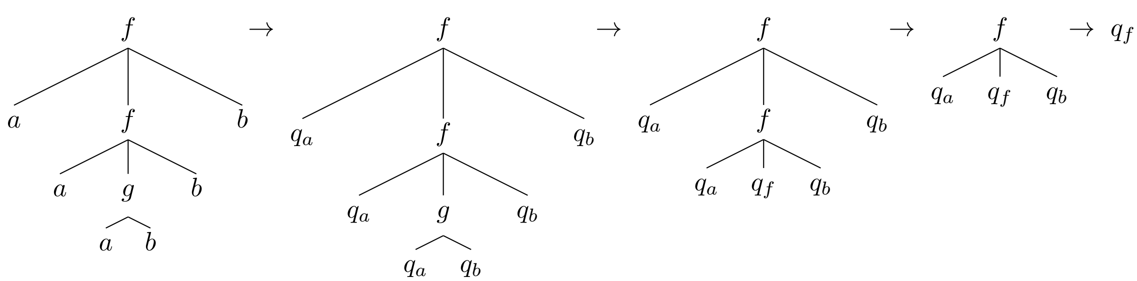
Beispiellauf:



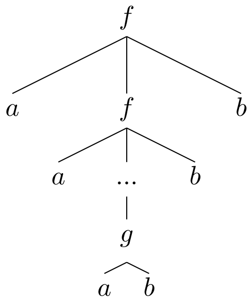
2. $(a^n b^n light)$

Betrachten $\mathcal{A} = (\{q_a, q_b, q_f\}, \{a^{(0)}, b^{(0)}, f^{(3)}, g^{(2)}\}, \{q_f\}, \Delta)$
mit $\Delta =$
$\{a \to q_a, b \to q_b, g(q_a, q_b) \to q_f, f(q_a, q_f, q_b) \to q_f\}$

Beispiellauf:



$\mathcal{A}$ akzeptiert also alle Bäume der Form:



3. Simulation eines Wortautomaten: (siehe Übung)

Betrachtet man $\Sigma = \{a^{(0)}, f^{(2)}, g^{(1)}\}$. Dann ist $L = \{f(g^i(a), g^i(a)) | i \geq 0\}$ nicht erkennbar.

1.10 Determinismus

Ein Automat $\mathcal{A}(Q, \Sigma, F, \Delta)$ heißt deterministisch, falls: aus $f(q_1, \ldots, q_n) \to q$ und $f(q_1, \ldots, q_n) \to q'$ folgt $q = q'$

1.11 Satz

Sei $\mathcal{A} = (Q, \Sigma, F, \Delta)$ ein Baumautomat, dann existiert ein deterministischer Baumautomat $\mathcal{A}_d$, so dass $L(\mathcal{A}) = L(\mathcal{A}_d)$.

Beweis: Setze $\mathcal{A}_d = (Q_d, \Sigma, F_d, \Delta_d)$
mit $Q_d = 2^Q$ $(*)$
und $f(s_1, \ldots, s_n) \to s \in \Delta_d$
$\Leftrightarrow s = \{q \in Q | \exists q_1 \in s_1 \ldots q_n \in s_n : (f(q_1, \ldots, q_n) \to q) \in \Delta\}$
und $F_d = \{s \in Q_d | s \cap F \neq \emptyset\}$.

Wir zeigen:
1. $\mathcal{A}$ ist deterministisch
2. $L(\mathcal{A}) \subset L(\mathcal{A}_d)$
3. $L(\mathcal{A}_d) \subset L(\mathcal{A})$

1. ist klar, denn $(*)$ ist mit einer Äquivalenz definiert.
2. „$L(\mathcal{A}) \subseteq L(\mathcal{A}_d)$“:

Wir zeigen hierzu: Ist $Z = \{q | t \to_{\mathcal{A}}^* q\}$, dann $t \to_{\mathcal{A}_d}^* z$.

Induktionsannahme:
Angenommen $a \to_{\mathcal{A}} q_a$, dann ist $q_a \in \{q \in Q | q \to_{\mathcal{A}}^* q\}$, das heißt

$a \to_{\mathcal{A}}^* q_a \Leftrightarrow q_a \in \{q \in Q | a \to_{\mathcal{A}}^* q\}$
$\Leftrightarrow q_a \in \{q \in Q | (a \to q) \in \Delta\}$
also $a \to_{\mathcal{A}}^* q_a \Leftrightarrow q_a \in \{q \in Q | (a \to q) \in \Delta\}$, das heißt
$z := \{q_a \in Q | a \to_{\mathcal{A}}^* q_a\} = \{q \in Q | (a \to q) \in \Delta\} =: s$

Nun ist $(a \to s) \in \Delta_d$ per Definition, also auch $(a \to z) \in \Delta_d$, damit: $a \to_{\mathcal{A}_d}^* z$.

Betrachten wir nun $t = \sigma(t_1, \ldots, t_n)$

Induktionsvoraussetzung:
$t_i \to_{\mathcal{A}_d}^* z_i$ mit $Z_i = \{q \in Q | t_i \to_{\mathcal{A}}^* q\}$
Das heißt, es existieren Läufe r_i für t_i von $\mathcal{A}_d$ mit $r_i(\varepsilon) = z_i$

Induktionsschritt:
zu zeigen: $t \to_{\mathcal{A}}^* z$ mit $Z = \{q \in Q | t \to_{\mathcal{A}}^* q\}$
Das heißt, es existiert ein Lauf r für t von $\mathcal{A}_d$ mit $r(\varepsilon) = z$
Das heißt, $\exists r$:

- $r(\varepsilon) = z$
- $r(i) = z_i$
- $\sigma(z_1, \ldots, z_n) \to z \in \Delta_d$

Setze nun $r_{|i} = r_i$, damit ist insbesondere $r(i) = r_i(\varepsilon) = z := \{q | t_i \to_{\mathcal{A}_d}^* q\}$

Es bleibt also zu zeigen: $\exists$ Regel $\sigma(z_i, \ldots, z_m) \to z \in \Delta_d$.
Es ist nun $z \in Z \Leftrightarrow t \to_{\mathcal{A}}^* z$
$\Leftrightarrow \exists q_i \in Q : t_i \to_{\mathcal{A}}^* q_i, \sigma(q_1, \ldots, q_m) \to z \in \Delta$
$\Leftrightarrow \exists z_i \in Z_i$ und $\sigma/z_1, \ldots, z_m) \to z \in \Delta$
Also $Z = \{z \in Q | \exists z_i \in Z_i : (\sigma/z_1, \ldots, z_m) \to z) \in \Delta)$ also per Definition $\sigma/z_1, \ldots, z_m) \to z \in \Delta_d$

2. „$L(\mathcal{A}_d) \subseteq L(\mathcal{A})$“:

Sei $t \in T_\Sigma$ mit $t \notin L(\mathcal{A})$, dann ist $Z \cap F = \{q \in Q | t \to_{\mathcal{A}}^* q\} \cap F = \emptyset$
Laut 2. ist $t \to_{\mathcal{A}_d}^* z$ (und $\mathcal{A}$ ist deterministisch) Wegen $Z \cap F = \emptyset$ ist $Z \notin F_d$, also $t \notin L(\mathcal{A}_d)$

Wir vereinbaren die Abkürzungen: NBA/NTA für nichtdeterministischer Baumautomat und DBA/DTA für deterministischer Baumautomat.
Wie im Wortfall ist die Konstruktion exponentiell, das heißt wir benötigen expontntiell viele Zustände ($Q_d = 2^{|Q|}$). Und wie im Wortfall lässt sich das im Allgemeinen nicht vermeiden.

Beispiel: Betrachtet man $\Sigma = \{f^{(1)}, g^{(1)}, a^{(0)}\}$ und sei $L_n = \{f \in T_\Sigma | t(\underbrace{1 \ldots 1}_{\text{n-mal}}) = f\}$

Ein NTA benötigt $n+2$ Zustände:
$\mathcal{A} = (Q, \Sigma, F, \Delta)$ mit $Q = \{q, q_1, \dots, q_{n+1}\}$, $F = \{q_{n+1}\}$
mit Übergängen $\Delta = \{a \to q, f(q) \to q, g(q) \to q, f(q) \to q_1,$
$f(q_i) \to q_{i+1}, g(q_i) \to q_{i+1}\}$ für $i \in \{1, \dots, n\}$

Man kann zeigen: Ein DTA $\mathcal{A}'$ mit $L(\mathcal{A}') = L_n$ hat mindestens 2^{n+1} Zustände.

1.12 Definition vollständing und reduziert

Ein Automat ($\mathcal{A} = Q, \Sigma, F, \Delta$) heißt:

- vollständig, falls für jedes $f^{(n)} \in \Sigma$ und alle $q_1, \dots, q_n \in Q$ eine Regel $f(q_1, \dots, q_n) \to q \in \Delta$ existiert.
- reduziert, falls für jeden Zustand $q \in Q$ ein Term $t \in T_\Sigma$ exisitert mit $f \to^*_{\mathcal{A}} q$

1.13 Satz

Sei $\mathcal{A}$ ein Baumautomat. Dann existiert ein vollständiger, reduzierter Baumautomat $\mathcal{A}'$ mit $L(\mathcal{A}) = L(\mathcal{A}')$.

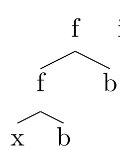
Für Wortautomaten gibt es das Pumping-Lemma, das die Gedächtnislosigkeit der Automaten formalisiert. Formal besagt es: Ist L eine reguläre Wortsprache, dann existiert ein $n \in \mathbb{N}$, so dass sich $w \in L$ mit $|w| > n$ zerlegen lässt in $w = xyz$, $y \neq \varepsilon$ und $\forall i \geq 0$ ist $xy^iz \in L$.

Baumautomaten haben auch kein Gedächtnis, also erwarten wir ein analoges Resultat. Dazu müssen wir formalisierten, was „aufgepumpt “werden soll.

1.14 Definition Kontext

Es sei Σ ein Rangalphabet und $x^{(0)} \notin \Sigma$. Es sei $C \in T_{\Sigma \cup \{x\}}$. Falls es genau eine Position $p \in Pos(C)$ gibt mit $C(p) = x$, dann heißt C ein Kontext.

Beispiel:

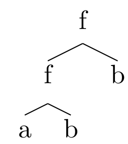


ist ein Kontext.

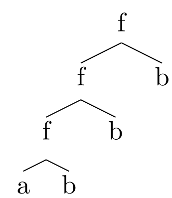
Wir schreiben $T_\Sigma(x)$ für die Menge aller solcher Kontexte.

Ist $C \in T_\Sigma(x)$ mit $C(p) = x$, dann schreiben wir $C[u]$ statt $C[u]_p$ für den Baum, der entsteht, wenn wir x durch u ersetzen.
Wir schreiben $C^0 = x$, $C^1 = C$, $C^n = C^{n-1}[C]$

Beispiel: Betrachtet $t =$



Setze $u = f(a,b)$ und $C = f(x,b)$.
Dann ist $t = C[u]$ und $C^2[u] =$

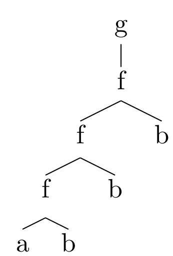


1.15 Pumping-Lemma

Sei $L \subseteq t_\Sigma$ erkennbar, dann existiert ein $k \in \mathbb{N}$, so dass:
Für alle $T \in L$ mit $ht(t) > k$ gibt es einen Kontext $C \in T_\Sigma(x)$, einen nicht-trivialen Kontext $C' \in T_\Sigma(x)$ und einen Term $u \in T_\Sigma$ mit $t = C[C'[u]]$ und $C[(C')^n[u]] \in L$ für alle $n \geq 0$.

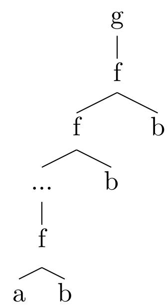
Beweis: Sei L erkennbar, das heißt $\exists$ Baumautomat $\mathcal{A} = (Q, \Sigma, F, \Delta)$ mit $L = L(\mathcal{A})$. Setze $|Q| = k$ und betrachte $t \in L$ mit $ht(t) > k$. Betrachte nun einen Lauf r und einen Pfad in t, der länger als k ist. Nun gibt es $p_1, p_2 \in Pos(r)$ mit $r(p_1) = r(p_2) = q \in Q$. Sei nun $u = t_{|p_2}$ der Sub-Baum von t bei p_2 und $u' = t_{|p_1}$. Dann existiert C' mit $C'[u] = u'$ und es existiert C mit $t = C[C'[u]]$. Es ist wegen $t \in L$
$C[C'[u]] \rightarrow^*_\mathcal{A} C[C'[q]] \rightarrow^*_\mathcal{A} C[q] \rightarrow^*_\mathcal{A} q_f \in F$, also auch
$C[(C')^n[u]] \rightarrow^*_\mathcal{A} C[(C')^n[q]] \rightarrow^*_\mathcal{A} CC[(C')^{(n-1)}[q]] \rightarrow^*_\mathcal{A} \cdots \rightarrow^*_\mathcal{A} C[q] \rightarrow^*_\mathcal{A} q_f \in F$. q.e.d.

Beispiel: Betrachte den Baumautomaten $\mathcal{A} = (\{q_a, q_b, q_g, q_f\}, \{a^{(0)}, b^{(0)}, g^{(1)}, f^{(2)}\}, \{q_f\}, \Delta)$ mit
$\Delta = \{a \rightarrow q_a, b \rightarrow q_b, f(q_a, q_b) \rightarrow q_g, g(q_g) \rightarrow q_f\}$



$u = f(a,b)$, $u' = C'[u] = f(f(a,b),b)$
$C = g(f(x,b))$, $C' = f(x,b)$

$C[(C')^n[u]] =$



Die Sprache $L = \{f(g^i(a), g^i(a)|i \geq 0\}$ kann nicht erkennbar sein, denn für große i würde man ein k finden, so dass ein gegebener Baumautomat auch $f(g^{i+lk}(a), g^i(a))$ für alle $l \geq 0$ akzeptiert.

1.16 Korollar

Für $\mathcal{A} = (Q, \Sigma, F, \Delta)$ ist $L(\mathcal{A}) \neq \emptyset \Leftrightarrow \exists t \in L$ mit $ht(t) \leq |Q|$:

- $L(\mathcal{A})|$ nicht endlich $\Leftrightarrow \exists t \in L$ mit $|Q| < ht(t) \leq 2|Q|$