HA-Lösung

TA-Lösung

Einführung in die theoretische Informatik – Aufgabenblatt 5

Beachten Sie: Soweit nicht explizit angegeben, sind Ergebnisse stets zu begründen!

Hausaufgaben: Abgabe bis zum 25.05.2016 (Mittwoch) um 12:00

Aufgabe 5.1 Levenshtein-Distanz **1P+3P+2P+2P**

Wir konstruieren ein Programm das Wörter einer regulären Sprache erkennt. Leider kann es passieren, dass der Benutzer kleine Fehler bei der Eingabe macht. Das System soll diese Fehler erkennen und einen Korrekturvorschlag machen.

Wir verwenden hierfür die Levenshtein-Distanz (auch: Edit-Distanz):

Um ein Wort w in ein anderes Wort w' zu überführen, erlaubt man die Operationen *replace (R)* $(a_1 \dots a_{i-1}a_ia_{i+1} \dots a_l \to a_1 \dots a_{i-1}ba_{i+1} \dots a_l)$, *delete (D)* $(a_1 \dots a_{i-1}a_ia_{i+1} \dots a_l \to a_1 \dots a_{i-1}\varepsilon a_{i+1} \dots a_l)$ und *insert (I)* $(a_1 \dots a_{i-1}a_ia_{i+1} \dots a_l \to a_1 \dots a_{i-1}a_iba_{i+1} \dots a_l)$ mit $a_i, b \in \Sigma$. Die Levenshtein-Distanz (geschrieben $\Delta(w, w')$) zwischen w und w' ist dann die minimale Anzahl von Operationen R, D, I, um w in w' umzuschreiben.

Wir schreiben $\Delta_{L,i} = \{w \in \Sigma^* \mid \exists w' \in L.\, \Delta(w', w) \leq i\}$ für die Sprache alle Wörter mit Edit-Distanz kleiner-gleich i zu L.

(a) Bestimmen Sie $\Delta(abcde, accd)$.

(b) Zeigen Sie: Wenn L regulär ist, dann ist auch $\Delta_{L,n}$ regulär. Konstruieren Sie hierzu einen ε-NFA N für $\Delta_{L,n}$ aus dem DFA für L. Sie müssen ausnahmsweise nicht die Korrektheit Ihres Verfahrens zeigen. Hinweis: Es ist hilfreich sich die Anzahl der bereits aufgetretenen Fehler in den Zuständen zu speichern.

(c) Beschreiben Sie ein Verfahren mit dem für ein Wort $w \in \Delta_{L,i}$ bestimmt werden kann, welches $w' \in L$ die kleinste Distanz zu w hat. Verwenden Sie hierzu die spezielle Struktur des konstruierten ε-NFAs N für $\Delta_{L,i}$.

(d) Sei nun $L = L((ab)^* \mid b^*)$. Konstruieren Sie den ε-NFA N für $\Delta_{L,1}$ und bestimmen Sie mithilfe von N alle Wörter w mit $\Delta(b, w) = 1$.

Lösung:

(a) $\Delta(abcde, accd) = 2$.

(b) Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA für L. Wir erhalten einen ε-NFA N für $\Delta_{L,n}$, in dem wir n Fehlerebenen einführen. Der Automat darf nicht-deterministisch einen Fehler machen muss dann aber zu einer höheren Fehlerebene wechseln. Formal:

$$N = (Q \times [0, n], \Sigma, \delta', (q_0, 0), F \times [0, n])$$

mit

$$\begin{aligned}
\delta' &= \{((q, i), a, (p, i)) \mid q, p \in Q \wedge i \leq n \wedge a \in \Sigma \wedge \delta(q, a) = p\} && \text{kein Fehler} \\
&\cup \{((q, i), \varepsilon, (p, i + 1)) \mid q, p \in Q \wedge i < n \wedge (\exists a \in \Sigma.\, \delta(q, a) = p)\} && \text{Delete} \\
&\cup \{((q, i), a, (q, i + 1)) \mid q \in Q \wedge i < n \wedge a \in \Sigma\} && \text{Insert} \\
&\cup \{((q, i), b, (p, i + 1)) \mid q, p \in Q \wedge i < n \wedge (\exists a \in \Sigma \setminus \{b\}.\, \delta(q, a) = p)\} && \text{Replace}
\end{aligned}$$

(c) Anhand der bekannten Algorithmen berechnen wir $\delta'((q_0, 0), w) \cap F \times [0, n]$ und wählen ein (q, i) mit minimaler Fehlerebene i. Wir betrachten nun einen akzeptierenden Lauf für w der in (q, i) endet. Immer wenn eine "Fehlerkante" benutzt wird, ändern wir das Wort w zu w' ab, so dass der Lauf in Ebene 0 in den entsprechenden Zielzustand wechseln kann.

(d) Wörter w mit $\Delta(b, w) = 1$: $\varepsilon, a, ab, ba, bb$.

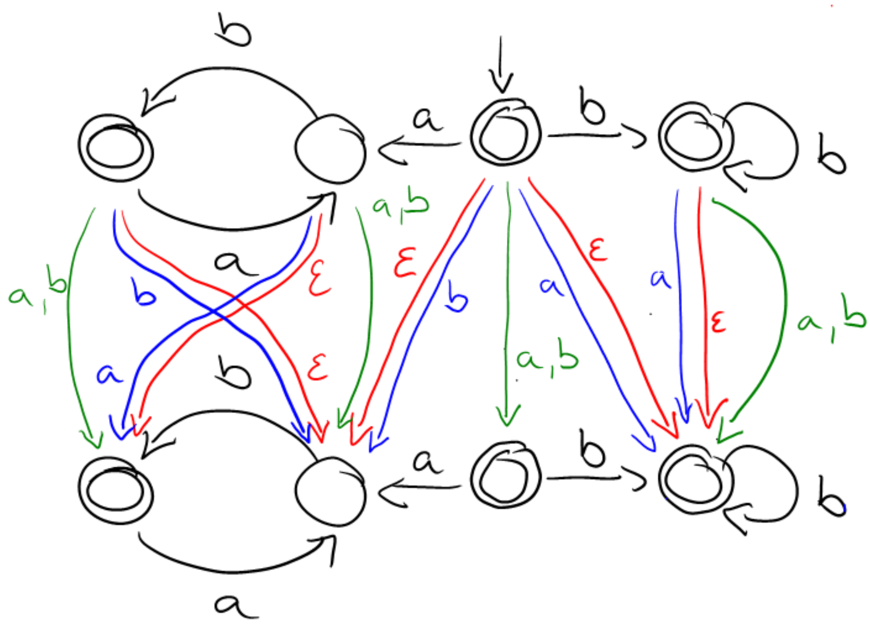


Abbildung 1: Konstruierter ε-NFA für L. Delete: rot, Insert: grün, Replace: blau.

Aufgabe 5.2 Leere Sprachen **2P**

Sie haben in der Vorlesung gesehen, wie man entscheiden kann, ob ein FA die leere Sprache akzeptiert. Wir entwickeln jetzt ein Verfahren, das direkt und ohne Umwege über FAs prüft, ob ein regulärer Ausdruck die leere Sprache beschreibt.

Definieren Sie hierzu eine rekursive Funktion $\mathsf{iszero} : RE \mapsto \mathbb{B}$ über den regulären Ausdrücken und beweisen Sie, dass für alle Ausdrücke r gilt:

$$\mathsf{iszero}\,(r) \Leftrightarrow L(r) = \emptyset$$

Lösung: **Konstruktion:** Wir definieren folgende rekursive Prozedur:

$$\begin{aligned} \mathsf{iszero}\,(\emptyset) &= \mathbf{true} & \mathsf{iszero}\,(\alpha\beta) &= \mathsf{iszero}\,(\alpha) \vee \mathsf{iszero}\,(\beta) \\ \mathsf{iszero}\,(\epsilon) &= \mathbf{false} & \mathsf{iszero}\,(\alpha \mid \beta) &= \mathsf{iszero}\,(\alpha) \wedge \mathsf{iszero}\,(\beta) \\ \mathsf{iszero}\,(a) &= \mathbf{false} & \mathsf{iszero}\,(\alpha^*) &= \mathbf{false} \end{aligned}$$

Korrektheit: Wir zeigen $\mathsf{iszero}\,(\gamma) \Leftrightarrow L(\gamma) = \emptyset$ mit struktureller Induktion über γ:

- $\gamma = \emptyset$: Trivial. • $\gamma = \epsilon$: Trivial. • $\gamma = a$: Trivial.
- $\gamma = \alpha\beta$: $\mathsf{iszero}\,(\alpha\beta) \Leftrightarrow \mathsf{iszero}\,(\alpha) \vee \mathsf{iszero}\,(\beta) \overset{\textbf{IH}}{\Leftrightarrow} L(\alpha) = \emptyset \vee L(\beta) = \emptyset \Leftrightarrow L(\alpha\beta) = \emptyset$.
- $\gamma = \alpha \mid \beta$: $\mathsf{iszero}\,(\alpha \mid \beta) \Leftrightarrow \mathsf{iszero}\,(\alpha) \wedge \mathsf{iszero}\,(\beta) \overset{\textbf{IH}}{\Leftrightarrow} L(\alpha) = \emptyset \wedge L(\beta) = \emptyset \Leftrightarrow L(\alpha|\beta) = \emptyset$.
- $\gamma = \alpha^*$: $\mathsf{iszero}\,(\alpha^*) \Leftrightarrow \mathbf{false} \Leftrightarrow L(\alpha^*) = \emptyset$

Aufgabe 5.3 Myhill-Nerode-Relation **1P+1P+1P+1P**

Bestimmen Sie zu folgenden Sprachen die Myhill-Nerode-Relation, indem Sie geeignet parametrisiert alle Äquivalenzklassen aufzählen.

(a) $L_a = \{w \in \{a,b\}^* \mid |w|_a \equiv 0 \pmod 2\}$

(b) $L_b = \{w \in \{a,b\}^* \mid |w|_a = 2\,|w|_b\}$

(c) $L_c = L((a^*b)^*)$

(d) $L_d = \{ww \mid w \in \{a,b\}^*\}$. Welche spezielle Eigenschaft haben die die Äquivalenzklassen: $[w]_{L_d}$?

Lösung:

(a) $[\varepsilon] = L_a$, $[a] = \{w \in \{a,b\}^* \mid |w|_a \equiv 1 \pmod 2\}$

(b) $[k] = \{w \in \{a,b\}^* \mid |w|_a - 2\,|w|_b = k\}$ für alle $k \in \mathbb{Z}$

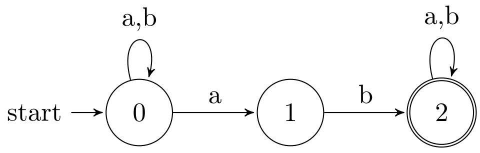
(c) $[\varepsilon] = L((a^*b)^*)$ und $[a] = L((a|b)^*a)$

(d) $[w] = \{w\}$ für alle $w \in \{a, b\}^*$. Alle Äquivalenzklassen enthalten genau ein Element.

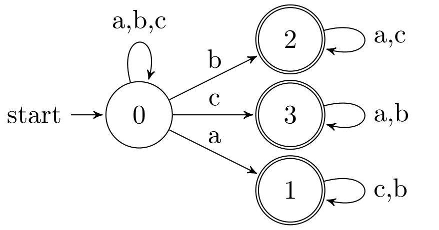
Aufgabe 5.4 Determinisieren + Minimieren **2P+2P+2P**

Determinisieren jeden der folgenden NFA und minimieren Sie dann jeweils den erhaltenen DFA. Konstruieren Sie bei der Determinisierung nur die vom Startzustand erreichbaren Zustände. Verwenden Sie für die Minimierung den erweiterten Algorithmus aus den Tutorübungen, um für jedes Paar von inäquivalenten Zuständen q, q' ein kürzestes Wort $w_{\{q,q'\}}$ mit $\delta(q, w_{\{q,q'\}}) \in F \leftrightarrow \delta(q', w_{\{q,q'\}}) \notin F$ zu berechnen.

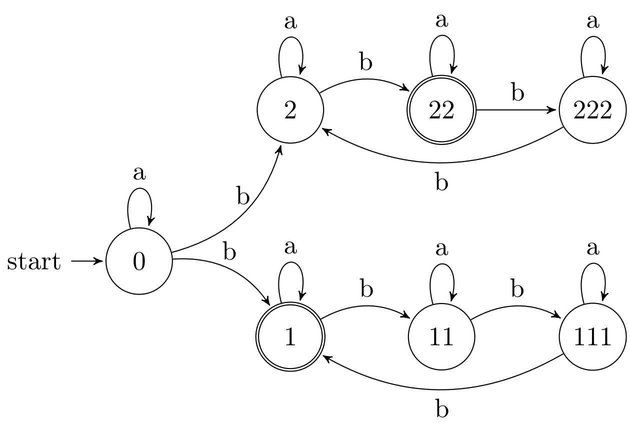
(a) NFA N_1:



(b) NFA N_2:

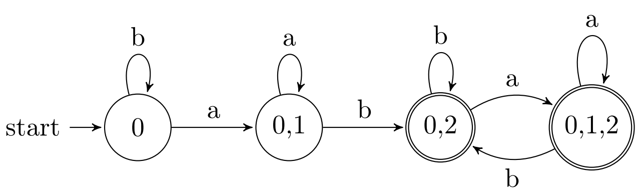


(c) NFA N_3:



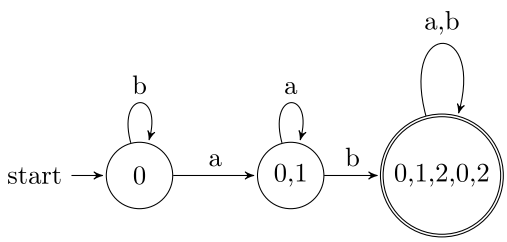
Lösung:

(a) DFA:

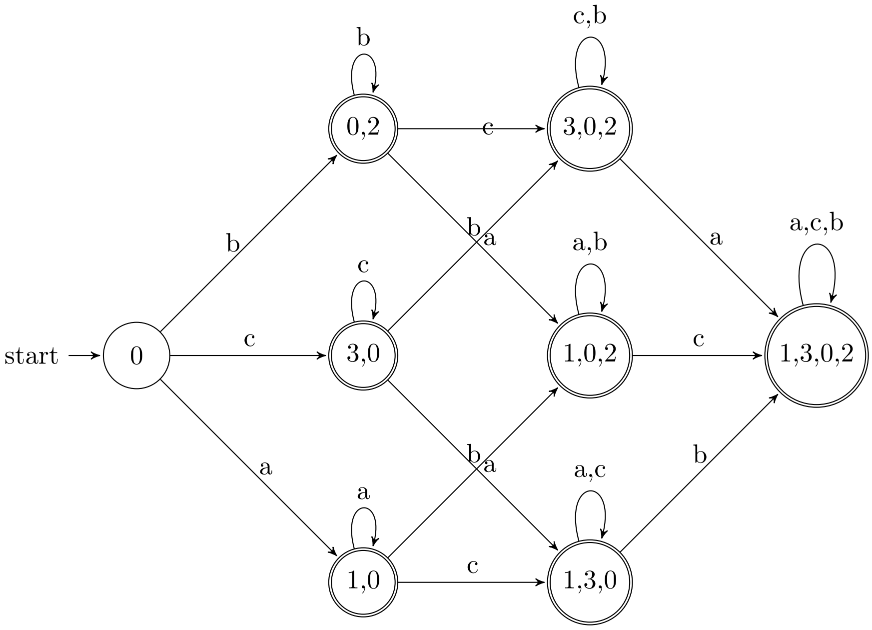


Minimierter DFA:

	0,1,2	0	0,2	0,1
0,1,2	–	–	–	–
0	ε	–	–	–
0,2	=	ε	–	–
0,1	ε	b	ε	–

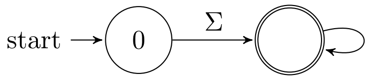


(b) DFA:

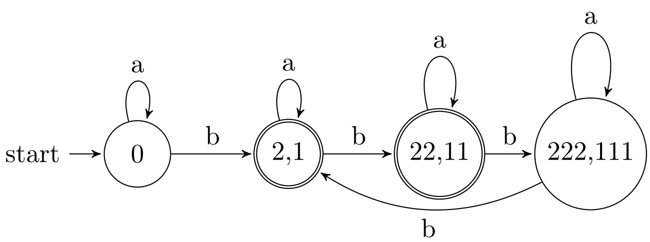


Minimierter DFA:

	1,0	3,0,2	0	1,3,0,2	0,2	1,3,0	1,0,2	3,0
1,0	–	–	–	–	–	–	–	–
3,0,2	=	–	–	–	–	–	–	–
0	ε	ε	–	–	–	–	–	–
1,3,0,2	=	=	ε	–	–	–	–	–
0,2	=	=	ε	=	–	–	–	–
1,3,0	=	=	ε	=	=	–	–	–
1,0,2	=	=	ε	=	=	=	–	–
3,0	=	=	ε	=	=	=	=	–

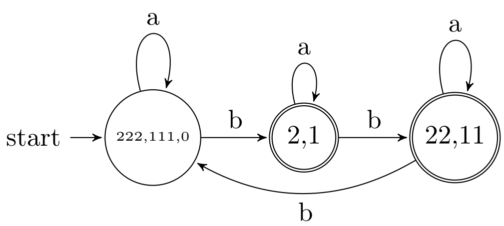


(c) DFA:



Minimierter DFA:

	2,1	22,11	0	222,111
2,1	–	–	–	–
22,11	b	–	–	–
0	ε	ε	–	–
222,111	ε	ε	=	–



Tutoraufgaben: Besprechung in KW20/21

Bitte beachten Sie die Regelung zu den Pfingstferien auf der Webseite.

Aufgabe 5.1

Mit $G = (\{S, E, O, A, B, X\}, \{a, b\}, P, S)$ sei die CFG mit folgenden Produktionen P bezeichnet:

$$\begin{array}{lcl} S & \to & E \mid O \\ E & \to & AB \mid BA \\ A & \to & XAX \mid a \\ B & \to & XBX \mid b \\ O & \to & XXO \mid X \\ X & \to & a \mid b \end{array}$$

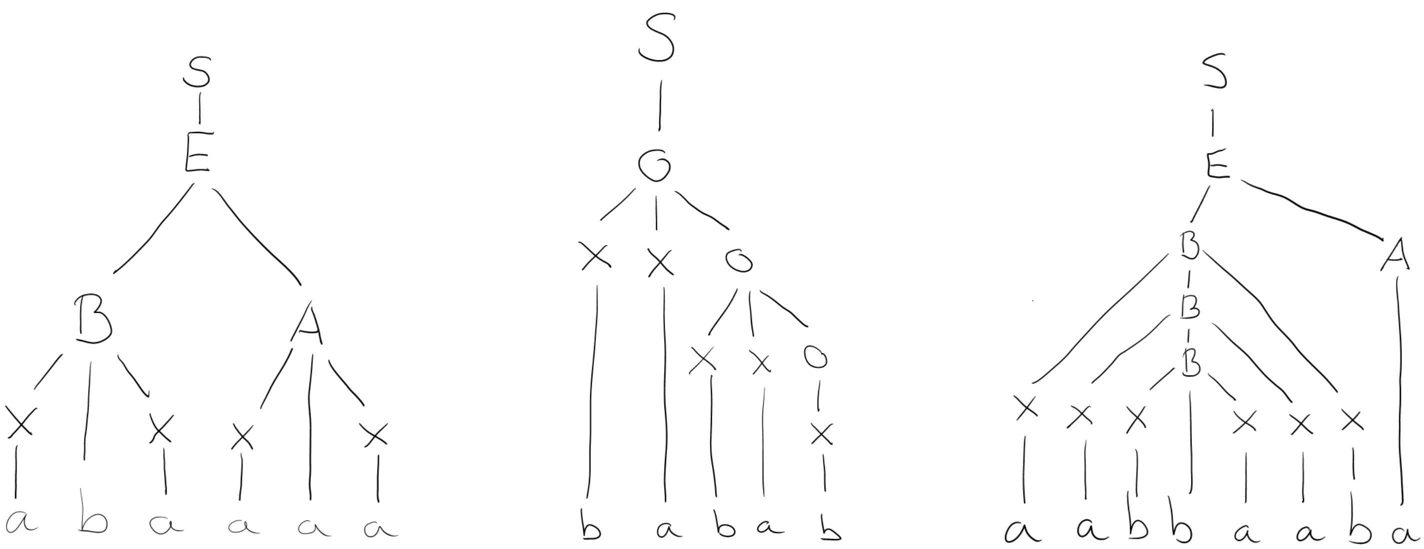
(a) Geben Sie für jedes der folgenden Wörter jeweils eine Linksableitung und eine Rechtsableitung zzgl. des entsprechenden Syntaxbaums an:

(i) *abaaaa* (ii) *babab* (iii) *aabbaaba*

(b) Bestimmen Sie die von $L(G)$ erzeugte Sprache und beweisen Sie, dass die Grammatik tatsächlich die angegebene Sprache erzeugt.

Lösung:

(a) Syntaxbäume:



Linksableitung für (i): $S \to E \to BA \to XbXA \to abXA \to abaA \to abaXaX \to abaaaX \to abaaaa$
Rechtsableitung für (i): $S \to E \to BA \to BXaX \to BXaa \to Baaa \to XbXaaa \to Xbaaaa \to abaaaa$

(b) Mit $L_Z(G)$ sei die von G erzeugte Sprache bezeichnet, wenn man Z als Startsymbol (Axiom) wählt.

Sei nun $L' = \bigcup_{k \geq 0} \{uav \mid u, v \in \Sigma^k\}$. Wir zeigen $L_A(G) = L'$.

(i) $L_A(G) \subseteq L'$.

▷ Strukturelle/wohlfundierte Induktion nach den Ableitungsregeln $A \to XAX$, $A \to a$. Sei $w \in L_A(G)$. Dann gibt es ein k, so dass w mit $3k + 1$ Ableitungsschritten abgeleitet worden ist. Dann $w \in \{uav \mid u, v \in \Sigma^k\}$.

$k = 0$: Dann wird w durch genau eine Produktionsanwendung erzeugt, d.h. w wird durch $A \to a$ erzeugt. Somit $w = a \in \{uav \mid u, v \in \Sigma^0\}$.

$k \to k + 1$: Es gilt $A \to XAX \to_G^* w = x_1 w' x_2$ mit $x_1, x_2 \in \{a, b\}$. Da gilt $A \to_G^{3k+1} w'$ und nach Induktion gilt $w' \in \{uav \mid u, v \in \Sigma^k\}$, so dass $w \in \{uav \mid u, v \in \Sigma^{k+1}\}$ folgt.

(ii) $L_A(G) \supseteq L'$

▷ Mittels Induktion nach k zeigen wir, dass jedes $w \in \{uav \mid u, v \in \Sigma^k\}$ eine Ableitung mit A als Startsymbol besitzt.

$k = 0$: Dann $w = a$ mit $A \to a = w$.

$k \to k + 1$: Dann $w = xw'y$ mit $x, y \in \{a, b\}$ und $w' \in \{uav \mid u, v \in \Sigma^k\}$. Nach Induktion besitzt w' eine Ableitung $A \to^* w'$, somit $A \to XAX \to Xw'X \to xw'X \to xw'y = w$.

Entsprechend folgt $L_B(G) = \bigcup_{k\geq 0}\{ubv \mid u,v \in \Sigma^k\}$ und $L_O(G) = \Sigma(\Sigma\Sigma)^*$, d.h. O produziert alle Wörter ungerader Länge.

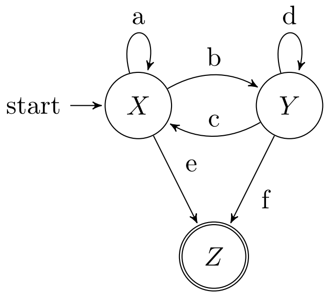
Für die restlichen Nichtterminale kann man nun direkt die erzeugten Sprachen angeben, da diese nur von Nichtterminalen abhängen, für welche die Sprachen schon bekannt sind:

$$L_E(G) = L_A(G)L_B(G) \cup L_B(G)L_A(G) = \bigcup_{i,j\geq 0} L((a|b)^i a(a|b)^{i+j} b(a|b)^j) \cup \bigcup_{i,j\geq 0} L((a|b)^i b(a|b)^{i+j} a(a|b)^j)$$

also die Sprache aller Wörter der Länge $2(i+j+1)$, die sich zumindest an den Positionen $i+1$ und $i+1+(i+j+1)$ unterscheiden. Da sich jedes Wort in $w \in L$ in einen der Fälle $|w|$ ungerade oder $|w| = 2n$ mit $w_k \neq w_{n+k}$ unterteillen lässt, folgt insgesamt, dass G die Sprache $\Sigma^* \setminus \{ww \mid w \in \Sigma^*\}$ erzeugt.

Aufgabe 5.2

Stellen Sie entsprechend der VL das zu folgendem FA gehörende Gleichungssystem auf. Bestimmen Sie dann mittels dem Gauß-Verfahren unter Verwendung von Ardens Lemma einen rationalen Ausdruck, der die vom FA akzeptierte Sprache beschreibt.



Lösung: Wir lesen folgendes Gleichungssystem ab:

$$\begin{array}{rcl} \{a\}X \cup \{b\}Y \cup \{e\}Z &=& X \\ \{c\}X \cup \{d\}Y \cup \{f\}Z &=& Y \\ \{\varepsilon\} &=& Z \end{array}$$

Lösen durch Auflösen nach Variablen mittels Ardens Lemma:

Für $A, B \subseteq \Sigma^*$ mit $\varepsilon \notin A$ hat die Gleichung

$$B \cup AX = X$$

eine und genau eine Lösung in 2^{Σ^*}, welche durch $X = A^*B$ gegeben ist.

Einsetzen von $Z = \{\varepsilon\}$

$$\begin{array}{rcl} \{a\}X \cup \{b\}Y \cup \{e\} &=& X \\ \{c\}X \cup \{d\}Y \cup \{f\} &=& Y \end{array}$$

Auflösen der Gleichung für Y nach Y (betrachte X als Konstante):

$$Y = \{d\}^*(\{c\}X \cup \{f\}) = L(d^*f) \cup L(d^*c)X$$

Einsetzen in Gleichung für X:

$$\{a\}X \cup \{b\}(\{d\}^*(\{c\}X \cup \{f\})) \cup \{e\} = (\{a\} \cup \{b\}\{d\}^*\{c\})X \cup \{b\}\{d\}^*\{f\} \cup \{e\} = L(a|bd^*c)X \cup L(e|bd^*f) = X$$

Auflösen nach X:

$$X = L((a|bd^*c)^*(e|bd^*f)$$

Aufgabe 5.3

Erinnerung: Ein Monoid $\langle M, \circ, 1\rangle$ besteht aus einer (Träger-)Menge M, einer assoziativen Abbildung $\circ\colon M \times M \to M$ und einem bzgl. $\circ$ neutralen Element $1 \in M$ (d.h. $\forall m \in M\colon m \circ 1 = m = 1 \circ m$). Ist $\circ$ kommutativ, dann wird $\langle M, \circ, 1\rangle$ als kommutatives Monoid bezeichnet. Gilt $\forall m \in M\colon m \circ m = m$, so wird $\langle M, \circ, 1\rangle$ als idempotentes Monoid bezeichnet.

Eine *Kleene Algebra* $\langle K, +, \cdot, ^*, 0, 1\rangle$ besteht aus einer Trägermenge K, den binären Operationen $+\colon K \times K \to K$ (Addition), $\cdot\colon K \times K \to K$ (Multiplikation), der unären Operation $^*\colon K \to K$ (Stern) und zweier nullärer Operationen/Konstanten $0, 1 \in K$. Mittels der Addition definiert man die binäre Relation $\sqsubseteq$ auf K durch

$$a \sqsubseteq b \overset{\text{Def}}{\longleftrightarrow} a + b = b$$

Wie üblich schreibt man kurz ab für $a \cdot b$. Um Klammern zu sparen, gilt "Stern vor Punkt vor Strich".

Eine Kleene Algebra erfüllt folgende Eigenschaften für alle $a, b, c, x \in K$:

Ax1: $\langle K, +, 0\rangle$ ist ein kommutatives und idempotentes Monoid.

Ax2: $\langle K, \cdot, 1\rangle$ ist ein Monoid.

Ax3: $a(b+c) = ab + ac$ und $(a+b)c = ac + bc$.

Ax4: $a0 = 0 = 0a$.

Ax5: $1 + aa^* \sqsubseteq a^*$ und $1 + a^*a \sqsubseteq a^*$.

Ax6: $b + ax \sqsubseteq x \to a^*b \sqsubseteq x$ und $b + xa \sqsubseteq x \to ba^* \sqsubseteq x$.

(a) Sei Σ ein Alphabet. Überprüfen Sie, dass $\langle 2^{\Sigma^*}, \cup, \circ, ^*, \emptyset, \{\varepsilon\}\rangle$ eine Kleene Algebra ist für

$$LL' := L \circ L' := \{ww' \mid w \in L, w' \in L'\} \qquad \text{und} \qquad L^* := \bigcup_{k \in \mathbb{N}_0} L^k$$

(b) Die übliche Addition und das Minimum auf $\mathbb{R}$ seien auf $[-\infty, \infty] = \mathbb{R} \cup \{\pm\infty\}$ wie folgt erweitert (mit $a \in \mathbb{R}$ beliebig):

$$\infty + a = \infty \qquad -\infty + a = -\infty \qquad -\infty + \infty = \infty \qquad \min(a, \infty) = a \qquad \min(a, -\infty) = -\infty \qquad \min(-\infty, \infty) = -\infty$$

Weiterhin gelte:

$$a^* := \begin{cases} -\infty & \text{falls } a \in [-\infty, 0) \\ 0 & \text{falls } a \in [0, \infty] \end{cases}$$

Überprüfen Sie wieder, dass $\langle \mathbb{R} \cup \{\pm\infty\}, \min, +, ^*, \infty, 0\rangle$ eine Kleene Algebra ist.

(c) Zeigen Sie, dass in jeder Kleene Algebra $\langle K, +, \cdot, ^*, 0, 1\rangle$ für beliebige $a, b, c, d, e, f \in K$ gilt:

(i) $\sqsubseteq$ ist eine partielle Ordnung auf K, die monoton bzgl. Addition, Multiplikation und Stern ist, d.h.:

$$a \sqsubseteq b \to (ca \sqsubseteq cb \wedge ac \sqsubseteq bc \wedge a + c \sqsubseteq b + c \wedge a^* \sqsubseteq b^*)$$

(ii) a^*b ist die bzgl. $\sqsubseteq$ kleinste Lösung der linearen Ungleichung $b + aX \sqsubseteq X$ in K (X Variable), genauer:

$$b + a(a^*b) = a^*b \qquad \wedge \qquad \forall x \in K \colon b + ax \sqsubseteq x \to a^*b \sqsubseteq x$$

Entsprechend ist ba^* die kleinste Lösung in K von $b + Xa \sqsubseteq X$.

(d) (i) (Knobelaufgabe) Zeigen Sie mittels Induktion nach n, dass jedes lineare Ungleichungsssystem

$$\begin{array}{rcl} a_{1,1}X_1 + a_{1,2}X_2 + \ldots a_{1,n}X_n + b_1 & \sqsubseteq & X_1 \\ & \vdots & \\ a_{n,1}X_1 + a_{n,2}X_2 + \ldots a_{n,n}X_n + b_n & \sqsubseteq & X_n \end{array}$$

mit $a_{i,j}, b_i \in K$ und $X_1, \ldots, X_n$ Variablen stets eine eindeutige $\sqsubseteq$-kleinste Lösung in K hat, d.h. dass es konkrete Elemente $x_1, \ldots, x_n \in K$ gibt, so dass für $X_1 = x_1, \ldots, X_n = x_n$ alle obigen Ungleichungen erfüllt sind, und für jede weitere Lösung $X_1 = y_1, \ldots, X_n = y_n \in K$ stets $x_i \sqsubseteq y_i$ gilt. Insbesondere gilt

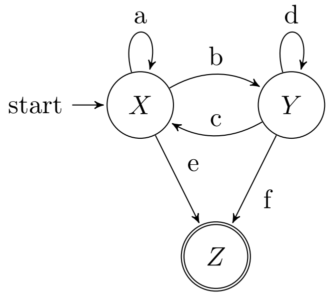
$$x_1 = a_{1,1}^*(a_{1,2}x_2 + \ldots + a_{1,n}x_n + b_1)$$

Machen Sie sich klar, dass man hiermit das Gauß-Verfahren zur Bestimmung von $x_1, \ldots, x_n$ verwenden kann.

(ii) Bestimmen Sie die kleinste Lösung x_1, x_2, x_3, x_4 für folgendes System:

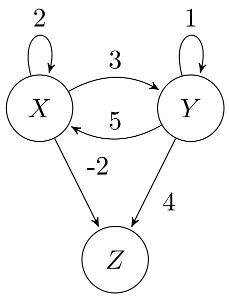
$$\begin{array}{rcl} aX + bY + eZ & \sqsubseteq & X \\ cX + dY + fZ & \sqsubseteq & Y \\ 1 & \sqsubseteq & Z \end{array}$$

(iii) Zeigen Sie, dass der für X in (5.2) berechnete Term gerade die Sprache $L(N)$ mit N der folgende NFA beschreibt, wenn man den Term über $\langle 2^{\Sigma^*}, \cup, \circ, ^*, \emptyset, \{\varepsilon\}\rangle$ auswertet:



(iv) Bestimmen Sie die Länge eines kürzesten Pfades von jedem Knoten zum Knoten Z in folgendem gewichteten Graphen.

Durch welche Werte muss man a, b, c, d, e, f konkret ersetzen, damit sich die in (d-ii) berechneten Terme in $\langle \mathbb{R} \cup \{\pm\infty\}, \min, +, \infty, 0\rangle$ zu den gesuchten Pfadlängen auswerten?



Lösung:

(a) Nur Stern überprüfen, Rest sollte klar sein. $\sqsubseteq$ entspricht hier $\subseteq$.

$1 + aa^* \sqsubseteq a^*$ wird zu $\{\varepsilon\} \cup AA^* \subseteq A^*$ mit $A \subseteq \Sigma^*$, was man leicht nachrechnet:

$$\{\varepsilon\} \cup AA^* = \{\varepsilon\} \cup A \bigcup_{k\geq 0} A^k = A^0 \cup \bigcup_{k\geq 1} A^k = A^*$$

Entsprechend für $1 + a^*a \sqsubseteq a^*$.

$b + ax \sqsubseteq x \to a^*b \sqsubseteq x$ wird zu

$$B \cup AX \subseteq X \to A^*B \subseteq X$$

mit $A, B, X \subseteq \Sigma^*$.

Mittels Induktion zeigt man nun, dass $A^kB \subseteq X$ für alle $k \in \mathbb{N}_0$ unter Verwendung von $B \cup AX \subseteq X$:

- $k = 0$: $A^0B = \{\varepsilon\}B = B \subseteq B \cup AX \subseteq X$.
- $k \to k+1$: Für $k \in \mathbb{N}_0$ beliebig fixiert gelte $A^kB \subseteq X$. Damit:

$$A^{k+1}B = A(A^kB) \subseteq AX \subseteq B \cup AX \subseteq X$$

Damit:

$$A^*B = \bigcup_{k\geq 0} A^kB \subseteq \bigcup_{k\geq 0} X = X$$

was zu zeigen war.

(b) Zur Verdeutlichung, welche Addition gemeint ist: $+_\mathbb{R}$ bezeichnet die übliche Addition auf den reellen Zahlen (erweitert auf $\pm\infty$).

Beachten: $\sqsubseteq$ auf K entspricht hier $\geq$ auf den reellen Zahlen

$$a \sqsubseteq b \text{ gdw. } \min(a, b) = b \text{ gdw. } a \geq b$$

$1 + aa^* \sqsubseteq a^*$ wird zu $\min(0, a +_\mathbb{R} a^*) \geq a^*$:

Gilt $a \geq 0$, so folgt $a^* = 0$ und damit $\min(0, a +_\mathbb{R} a^*) = 0 = a^*$; gilt $a < 0$, so folgt $a^* = -\infty$ und damit $\min(0, a +_\mathbb{R} a^*) = -\infty = a^*$. Für $1 +_\mathbb{R} a^*a \sqsubseteq a$ symmetrisch.

Aus $b + ax \sqsubseteq x \to a^*b \sqsubseteq x$ wird $\min(b, a +_\mathbb{R} x) \geq x \to a^* +_\mathbb{R} b \geq x$:

Aus $\min(b, a +_\mathbb{R} x) \geq x$ folgt $b \geq x \wedge a +_\mathbb{R} x \geq x$, damit $a \geq 0$ und somit auch $a^* = 0$, womit $a^* +_\mathbb{R} b \geq x$ stets gilt.

(c) (i) Reflexivität folgt aus der Definition von $\sqsubseteq$ und der Idempotenz von $+$:

$$a \sqsubseteq a \overset{\text{nDef}}{\longleftrightarrow} a + a = a$$

Antisymmetrie folgt aus der Definition von $\sqsubseteq$ und der Kommutativität von $+$:

$$b \overset{a\sqsubseteq b}{\equiv} a + b = b + a \overset{b\sqsubseteq a}{\equiv} a$$

Transitivität folgt aus der Definition von $\sqsubseteq$ und der Assoziativität von $+$:

$$c \overset{b\sqsubseteq c}{\equiv} c + b \overset{a\sqsubseteq b}{\equiv} c + (b + a) = (c + b) + a \overset{b\sqsubseteq c}{\equiv} c + a$$

Es gelte $a \sqsubseteq b$. Monotonie von $+$:

$$c + b \overset{a \sqsubseteq b}{=} c + (b + a) = (c + a) + b \overset{\text{Def} \sqsubseteq}{\sqsupseteq} c + a$$

Entsprechend Monotonie von $\cdot$:

$$cb \overset{a \sqsubseteq b}{=} c(b + a) = cb + ca \overset{\text{Def} \sqsubseteq}{\sqsupseteq} ca$$

Bleibt die Monotonie von *: mit Ax5 und der Monotonie von $\cdot$

$$b^* \sqsupseteq 1 + bb^* \sqsupseteq 1 + ab^*$$

womit sofort aus Ax6 $a^* \sqsubseteq b^*$ folgt (in Ax6 1 für b und b^* für x substituieren).

(ii) Wir zeigen zuerst den Speziallfall mit $b = 1$:

$1 + aa^* \sqsubseteq a^*$ ist Ax5. Bleibt $a^* \sqsubseteq 1 + aa^*$.

Nach Ax6 $b + ax \leq x \to a^* b \leq x$ mit 1 substituiert für b und $1 + aa^*$ substituiert für x reicht es zu zeigen, dass

$$1 + a(\underbrace{1 + aa^*}_{\sqsubseteq a^* \text{(Ax5)}}) \sqsubseteq 1 + aa^*$$

was aber wegen Ax5 ($1 + aa^* \sqsubseteq a^*$) und der Monotonie von $\sqsubseteq$ gilt. Symmetrisch folgt $1 + a^* a = a^*$.

Damit folgt auch sofort $b + a(a^* b) = (1 + aa^*)b = a^* b$ und symmetrisch $b + (ba^*)a = ba^*$.

Somit hat $b + aX \sqsubseteq X$ mit $X = a^* b$ mindestens eine Lösung, für – wie gerade gezeigt – die sogar Gleichheit gilt, nach Ax5 ist jede weitere Lösung größer als $a^* b$. Symmetrisch für $b + Xa \sqsubseteq X$ und $X = ba^*$.

(d) (i) Siehe Knobelaufgabe.

(ii) $1 \sqsubseteq Z$ führt auf $Z = 1$.

Damit ergibt sich $Y = d^*(cX + f)$ aus

$$cX + dY + fZ \sqsubseteq Y$$

Und damit $X = (a + bd^* c)^*(e + bd^* f)$ aus

$$aX + bY + eZ \sqsubseteq X$$

Damit schließlich $Y = d^* c(a + bd^* c)^*(e + bd^* f) + d^* f$.

Bemerkung: Man kann natürlich auch zuerst nach X und dann nach Y lösen, was auf

$$X = a^*(bY + e)$$

und damit auf

$$Y = (d + ca^* b)^*(f + ca^* e) \qquad X = a^* b(d + ca^* b)^*(f + ca^* e) + a^* e$$

führt.

Da die erhaltenen Terme jeweils die eindeutige kleinste Lösung beschreiben müssen, folgt z.B.

$$Y = (d + ca^* b)^*(f + ca^* e) = d^* c(a + bd^* c)^*(e + bd^* f) + d^* f$$

(iii) Stellt man das LGS nach VL für den NFA auf, so erhält man genau das System aus (c-ii) – mit der einzigen Aussnahme, dass statt der abstrakten Operationen die konkreten Operationen für Sprachen verwendet werden. Die Lösungsschritte aus (c-ii) können somit auch im konkreten Fall angewendet werden. Insbesondere beschreiben die Terme aus (c-i) bzw. die entsprechenden regulären Ausdrücke gerade alle Pfade von X bzw. Y nach Z (siehe auch Folien).

Beispiel: aus

$$d^* c(a + bd^* c)^*(bd^* f + e) + d^* f$$

wird

$$L(d^* c(a|bd^* c)^*(bd^* f|e)|d^* f)$$

(iv) Von X nach Z geht man direkt: -2

Von Y nach Z geht man über X: $5 - 2 = 3$

Von Z nach Z: 0

Setzt man $a = 2$, $b = 3$, $c = 5$, $d = 1$, $e = -2$ und $f = 4$, so werten sich die Ausdrücke aus (c-i) mit min als Addition und $+$ als Multiplikation gerade zu diesen Werten aus.

Beispiel: Aus

$$d^* c(a + bd^* c)^*(bd^* f + e) + d^* f$$

wird

$$\min(1^* + 5 + (\min(2, 3 + 1^* + 5))^* + \min(3 + 1^* + 4, -2), 1^* + 4) = \min(0 + 5 + 0 - 2, 0 + 4) = 3$$

Knobelaufgabe

Freiwillige Abgabe per E-Mail an theo-uebungsleitung@in.tum.de. Alle richtigen Einsendungen bis zum 25.05.2016 um 12:00 Uhr erhalten die Bonuspunkte. Wir bewerten nur die erste Einsendung und behalten uns das Recht vor unleserliche oder unverständliche Abgaben nicht zu korrigieren.

Aufgabe 5.1

Zeigen Sie die Behauptung aus TA5.3(c)(i)!