Prof. J. Esparza
S. Sickert, J. Krämer
Technische Universität München
KEINE ABGABE

Vielen Dank an Jan Wagener für die erweiterten Aufgabenlösungen

Einführung in die theoretische Informatik

Sommersemester 2017 – Übungsblatt 3

Übungsblatt

Wir unterscheiden zwischen Übungs- und Abgabeblättern. Auf diesem *Übungsblatt* finden Sie eine Übersicht über die Kernaspekte, die Sie in Kalenderwoche 20 in den Tutorien diskutieren, üben und vertiefen. Die Aufgaben auf diesem Blatt dienen dem Üben und Verstehen des Vorlesungsstoffes, sowie dem *eigenständigen Erarbeiten* der Kernaspekte. Außerdem sollen Ihnen diese Aufgaben auch helfen, ein Gefühl dafür zu bekommen, was Sie inhaltlich in der Klausur erwartet. Klausuraufgaben können jedoch deutlich von den hier gestellten Aufgaben abweichen. Abschreiben und Auswendiglernen von Lösungen wird Ihnen daher keinen dauerhaften Erfolg in der Vorlesung bringen. Fragen zu den Übungsblättern können Sie montags bis donnerstags von 12 Uhr bis 14 Uhr in der *THEO-Sprechstunde* in Raum 03.11.034 stellen.

Kernaspekte

K3.1 korrektes Wiedergeben der folgenden Definitionen

- regulärer Ausdruck
- sternfreier Ausdruck
- ε-NFA
- Produktkonstruktion

K3.2 eine (natürlichsprachlich) gegebene Sprache als regulären Ausdruck darstellen
K3.3 strukturelle Induktion über Grammatiken und reguläre Ausdrücke führen, um Eigenschaften der beschriebenen Sprache zu beweisen
K3.4 Funktionen über reguläre Ausdrücke definieren
K3.5 einen regulären Ausdruck r in einen NFA N übersetzen, so dass $\mathsf{L}(r) = \mathsf{L}(\mathsf{N})$
K3.6 einen ε-NFA N mittels Potenzmengenkonstruktion in einen DFA D übersetzen, so dass $\mathsf{L}(\mathsf{N}) = \mathsf{L}(\mathsf{D})$
K3.7 den Schnitt zweier regulärer Sprachen mithilfe der Produktkonstruktion bilden
K3.8 begründet entscheiden, ob gegebene Beispiele neu eingeführte Definitionen erfüllen
K3.9 Aussagen, mit neu eingeführten Definitionen beweisen oder widerlegen

Die folgende Aufgabe kann auch online gelöst werden. Melden Sie sich auf `http://automatatutor.com` *an und schreiben Sie sich in den Kurs* ***130THEO20*** *mit dem Passwort* ***IJD1DLKQ*** *ein. Wir empfehlen die Benutzung von Mozilla Firefox, da mit anderen Browsern die Darstellung teilweise inkorrekt sein kann.*

AUFGABE 3.1.

Stufe B

Geben Sie für jede der folgenden Sprachen einen regulären Ausdruck an, der genau die Sprache beschreibt. Verwenden Sie für die ersten drei Aufgaben das Alphabet $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$ und für die letzten beiden $\Sigma = \{0, 1\}$.

(a) Wörter gerader Länge.
(b) Wörter, die mit einem a beginnen und enden, sowie Wörter, die mit einem b beginnen und enden.
(c) Wörter, in denen kein a neben einem b steht.
(d) Zahlen in Binärdarstellung (most-significant-bit-first), die durch 2 teilbar sind.
(e) Zahlen in Binärdarstellung (most-significant-bit-first), die nicht durch 4 teilbar sind.

Erweiterte Lösungsskizze
Sei $\Sigma = \{\mathtt{a},\ b, \mathtt{c}\}$. Gib einen regulären Ausdruck für gegebene Sprache an.

(a) Wörter gerader Länge

$$\Big(\underbrace{(\mathtt{a}|\mathtt{b}|\mathtt{c})(\mathtt{a}|\mathtt{b}|\mathtt{c})}_{\text{Zwei beliebige Zeichen des Alphabets}} \Big)^{\overbrace{*}^{\text{Dies beliebig oft}}}$$

(b)
- Wörter, die mit einem a beginnen und enden, sowie
- Wörter, die mit einem b beginnen und enden

$$\underbrace{\mathtt{a}\ (\mathtt{a}|\mathtt{b}|\mathtt{c})^*\ \mathtt{a}}_{\text{Wörter, die mit a beginnen und enden}} \ \Big|\ \underbrace{\mathtt{b}\ (\mathtt{a}|\mathtt{b}|\mathtt{c})^*\ \mathtt{b}}_{\text{Wörter, die mit b beginnen und enden}} \ \Big|\ \underbrace{\mathtt{a} \mid \mathtt{b}}_{\text{Randfälle, die durch die ersten beiden Ausdrücke nicht abgedeckt sind}}$$

(c) Wörter, in denen kein a neben einem b steht

$$\Big(\underbrace{(\mathtt{a}^*|\mathtt{b}^*)}_{\text{Auf Ketten von a's oder b's..}} \quad \underbrace{\mathtt{cc}^*}_{\text{..folgt stets mindestens ein c..}} \Big)^* \ \underbrace{(\mathtt{a}^*|\mathtt{b}^*)}_{\text{..es sei denn sie stehen am Schluss.}}$$

$\Sigma = \{0, 1\}$

(d) Zahlen in Binärdarstellung, die durch 2 teilbar sind

$$(0|1)^*\ 0$$

⇝ Durch 2 teilbare Zahlen in Binärdarstellung enden stets auf '0'

(e) Zahlen in Binärdarstellung, die **nicht** durch 4 teilbar sind

- Ausführliche Version

$$(0|1)^*\ (01|11|10)\ \Big|\ \underbrace{1}_{\text{Randfall nicht vergessen!}}$$

⇝ Durch 4 teilbare Zahlen in Binärdarstellung enden stets auf '00' und nicht durch 4 teilbare Zahlen auf '01', '10' oder '11'.

- Kompaktere Version:

$$(0|1)^*\ (10\ |\ \underbrace{1}_{\text{Fast Fälle '01' und '11' als 'endet auf 1' zusammen}}\ )$$

⇝ Randfall '1' wird hiermit bereits abgedeckt, muss also nicht extra aufgeführt werden

AUFGABE 3.2. Stufe C

Sei $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. Geben Sie einen regulären Ausdruck mit möglichst wenigen Zeichen für die Sprache an, in der alle Wörter gleich oft die Zeichenketten ab und ba enthalten.
Beispiel: Das Wort abab enthält zweimal ab, aber nur einmal ba und soll somit *kein* Element der Sprache sein.

Erweiterte Lösungsskizze
Sei $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. Gib einen regulären Ausdruck für die Sprache aller Wörter an, die die Zeichenkette

ab

genauso oft enthalten, wie die Zeichenkette

ba.

Vorbemerkung: Betrachtet man nichtleere Wörter in der Sprache, so gilt es, zwei Fälle zu unterscheiden:

- Wörter, die mit einem a beginnen und
- Wörter, die mit einem b beginnen

Nimmt man nun erstere, so ist klar, dass alle Wörter der Form a^* in der Sprache enthalten sind (sie enthalten beide Zeichenketten 0-mal).
Erscheint allerdings das erste b, so tritt die Zeichenkette ab zum ersten Mal auf, d.h. Wörter der Form $\mathsf{a}^*\mathsf{b}$ liegen nicht in der Sprache, da sie 1-mal die Zeichenkette ab aber 0-mal die Zeichenkette ba enthalten.
Danach muss also irgendwann ein ba folgen. Dies ist definitiv vor dem nächsten (falls vorhanden) ab der Fall, da das bisherige Wort ja auf b endet. Nach diesem ba befinden wir uns wieder in der Ausgangssituation: Letztes bisheriges Zeichen ein a und gleichviele ab's wie ba's.
Induktiv gilt also: Ein mit a beginnendes Wort liegt in gegebener Sprache gdw. es auch auf a endet.
Und analog für b's: Ein mit b beginnendes Wort liegt in gegebener Sprache gdw. es auch auf b endet.
Wir haben es also mit einer ähnlichen Sprache, wie in Aufgabe 3.1b zu tun, nur jetzt über dem Alphabet $\{\mathsf{a},\mathsf{b}\}$ und mit ε.

Ausführliche Version: $\mathsf{a}\ (\mathsf{a}|\mathsf{b})^*\ \mathsf{a}\ \Big|\ \mathsf{b}\ (\mathsf{a}|\mathsf{b})^*\ \mathsf{b}\ \Big|\ \mathsf{a}|\mathsf{b}|\varepsilon$

$\rightsquigarrow$ Analog zu Aufgabe 3.1b

Komprimierte Version: $\underbrace{(\mathsf{ab}^*)^*\mathsf{a}}_{\text{Beginnt und endet auf a}}\ \Big|\ \underbrace{(\mathsf{ba}^*)^*\mathsf{b}}_{\text{Beginnt und endet auf b}}\ \Big|\ \varepsilon$

AUFGABE 3.3. Stufe C

Wir übersetzen den regulären Ausdruck $\mathsf{r} = (\mathsf{a} \mid \mathsf{b})^*\mathsf{a}$ in zwei Schritten zu einem DFA.

(a) Geben Sie **mit dem Verfahren aus der Vorlesung** einen ε-NFA N mit $\mathsf{L}(\mathsf{N}) = \mathsf{L}((\mathsf{a} \mid \mathsf{b})^*\mathsf{a})$ an. Halten Sie sich strikt an die Konstruktion aus den Folien. Als NFA für $\mathsf{x} \in \Sigma$ verwenden Sie hierfür den (bis auf Knotenbenennung) kanonischen ε-NFA $(\{\mathsf{q}_0, q_1\}, \{\mathsf{a}, \mathsf{b}\}, \{(\mathsf{q}_0, \mathsf{x}, q_1) \mid \mathsf{x} \in \Sigma\}, q_0, \{q_1\})$.

(b) Die Potenzmengenkonstruktion lässt sich auf ε-NFAs wie folgt erweitern:
Gegeben ein ε-NFA $\mathsf{N} = (\mathsf{Q}, \Sigma, \delta, \mathsf{q}_0, \mathsf{F})$ sei $\mathsf{D}' = (\mathsf{Q}', \Sigma', \delta', \mathsf{I}', \mathsf{F}')$ wie folgt definiert:

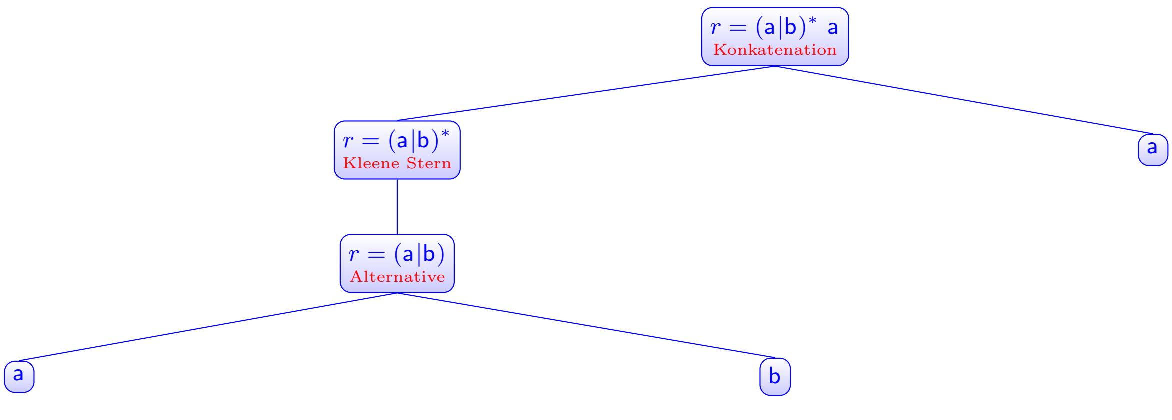
- $\mathsf{I}' := \bigcup_{i\geq 0} \delta(q_0, \varepsilon^i)$.
- Q' und $\delta' \colon \mathsf{Q}' \times \Sigma \to \mathsf{Q}'$ sind induktiv wie folgt definiert:
  - $\mathsf{I}' \in \mathsf{Q}'$.
  - Falls $S \in \mathsf{Q}'$, dann
  $$\delta'(S, \mathsf{x}) := \bigcup_{q\in S, i\in\mathbb{N}_0, j\in\mathbb{N}_0} \delta(q, \varepsilon^i \mathsf{x} \varepsilon^j)$$
  für jedes $\mathsf{x} \in \Sigma$ und $\delta'(S, \mathsf{x}) \in \mathsf{Q}'$.
  - Ansonsten enthält Q' keine weiteren Elemente.
- $\mathsf{F}' := \{S \in \mathsf{Q}' \mid S \cap \mathsf{F} \neq \emptyset\}$.

Determinisieren Sie mithilfe **dieser** Konstruktion den ε-NFA aus Aufgabenteil (a), um einen DFA mit gleicher Sprache zu erhalten.

Erweiterte Lösungsskizze
$r = (\mathsf{a}|\mathsf{b})^*\ \mathsf{a}$

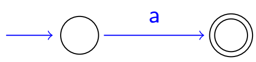
(a) Konstruiere einen ε-NFA N mit $\mathsf{L}(\mathsf{N}) = \mathsf{L}(r)$.

I: Zerlege r in seine Bestandteile (z.B. als Syntaxbaum)

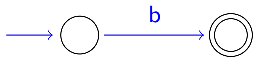


II: Konstruiere ε-NFAs für die (vorkommenden) Basisfälle ε, $\emptyset$, $\mathsf{a} \in \Sigma$

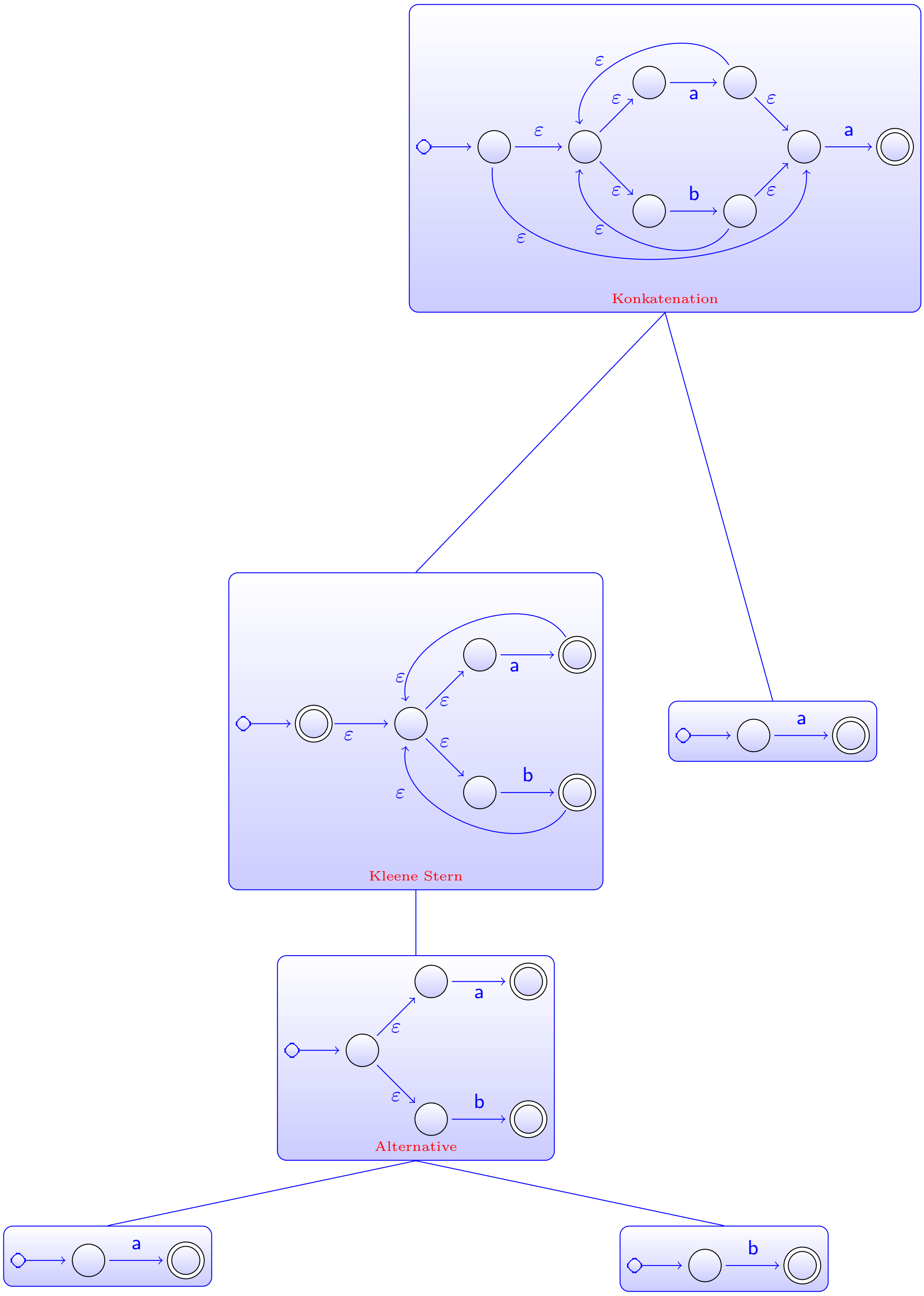
ε-NFA für die Sprache $\{\mathsf{a}\}$:



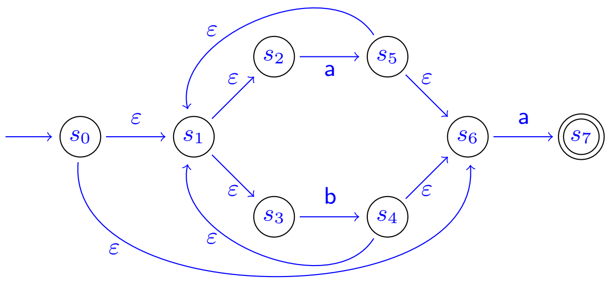
ε-NFA für die Sprache $\{\mathsf{b}\}$:



III: Füge die in II konstruierten Automaten in der in I festgelegten Reihenfolge mittels der Konstruktionen aus der Vorlesung (siehe Folien 59-62) zusammen

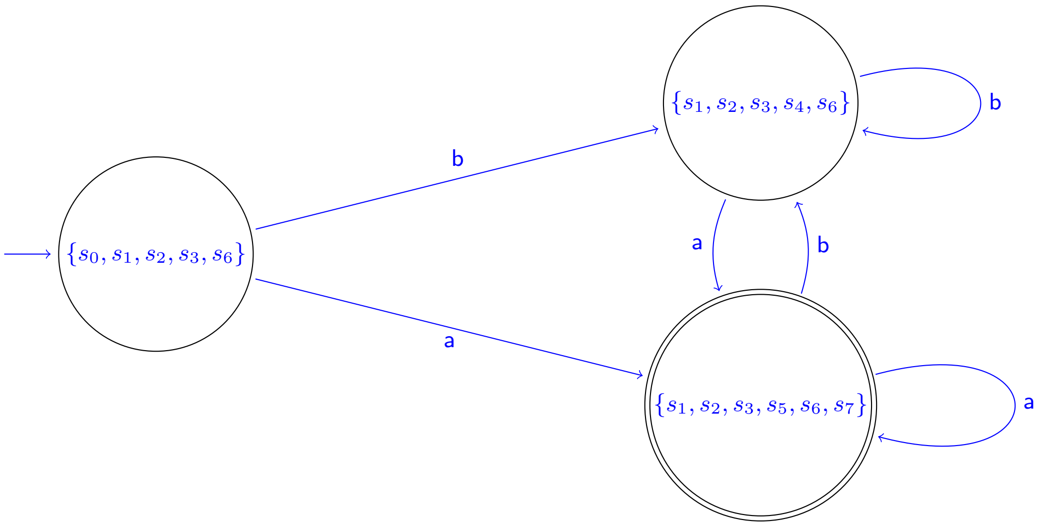


(b) Determinisiere N mittels (erweiterter) Potenzmengenkonstruktion



Vorgehen:

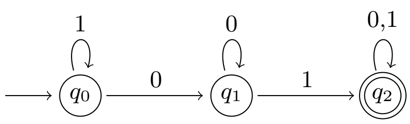
- Jeder Zustand des determinisierten Automaten D′ steht für (d.h. ßpeichert") die Menge an Zuständen aus N, in denen man sich mit dem bisher eingelesenen Wort gerade befinden könnte.
- Startzustand ist also die Menge an Zuständen aus N, in die man ohne Einlesen eines Zeichens, also allein durch Nutzung der ε-Kanten, gelangen kann.
- Endzustand ist ein Zustand in D′, wenn er mindestens einen Endzustand aus N enthält. Das bedeutet, wenn ein Wort dort landet, dann könnte man sich in N in einem Endzustand befinden, d.h. es gibt einen Lauf dieses Wortes in N, der in einem Endzustand endet.
- Transitionen von einem Zustand $S = \{s_{i_1}, ..., s_{i_n}\}$ mit Zeichen a im DFA D′ gehen in den Zustand, der für die Menge aller Zustände in N steht, die man durch Einlesen von genau einem a, also Nutzen genau einer mit a beschrifteter Kante und beliebig vieler ε-Kanten, von einem der s_{i_j} aus erreichen kann.

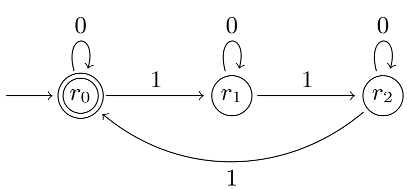


Somit erhält man:

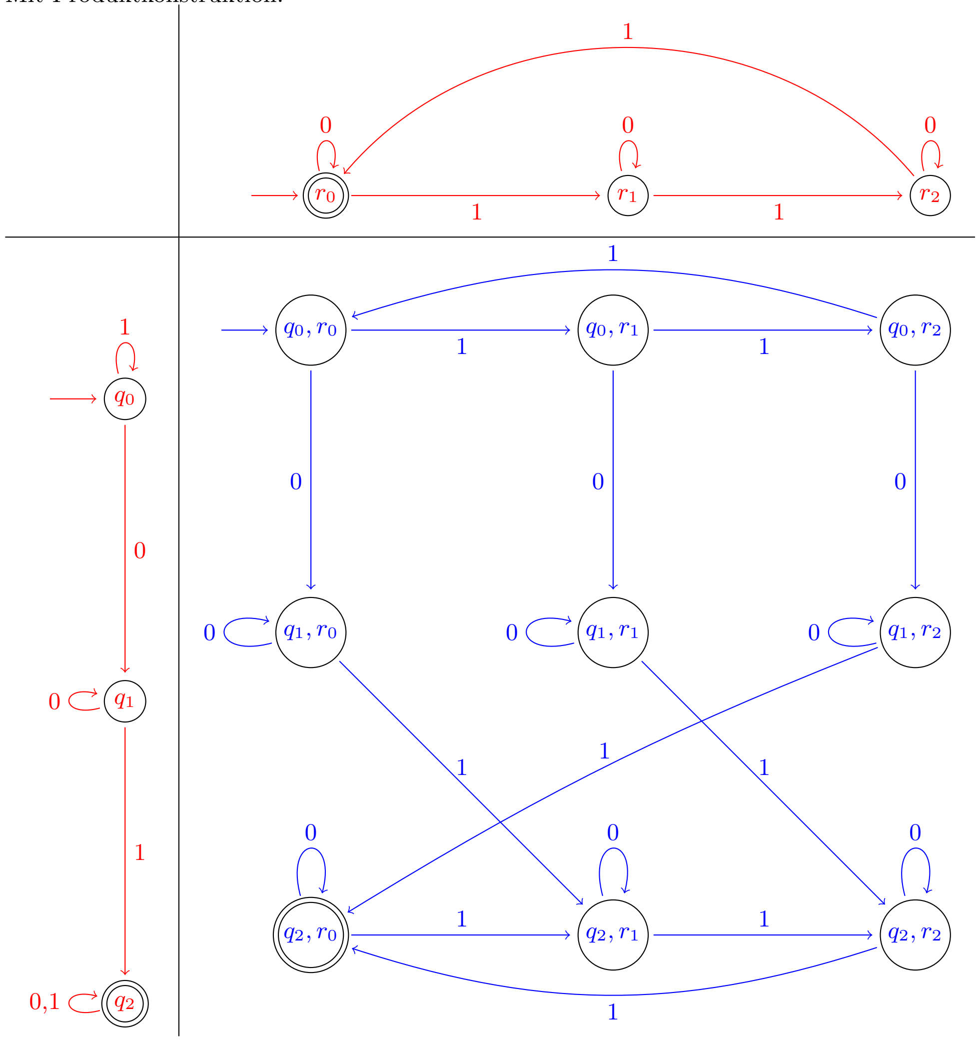
AUFGABE 3.4. Stufe C

Gegeben seien zwei DFAs D_1 und D_2 über dem gleichen Eingabealphabet $\Sigma = \{0, 1\}$. Verwenden Sie das Verfahren aus der Vorlesung um einen DFA D anzugeben, so dass $L(D) = L(D_1) \cap L(D_2)$.

D_1 : 

D_2 : 

Erweiterte Lösungsskizze
Sei $\Sigma = \{0, 1\}$. Konstruiere einen DFA D, der den Schnitt der Sprachen der gegebenen DFAs D_1 und D_2 akzeptiert, also:
$$\mathsf{L}(\mathsf{D}) = \mathsf{L}(\mathsf{D}_1) \cap \mathsf{L}(\mathsf{D}_2).$$
Mit Produktkonstruktion:



Definition (Suffix-Sprache)

Sei $\mathsf{L} \subseteq \Sigma^*$ eine Sprache über dem Alphabet Σ. Wir definieren $\mathsf{L}_{\mathsf{sf}} := \{v \in \Sigma^* \mid \exists u \in \Sigma^*.\ uv \in \mathsf{L}\}$ und bezeichnen L_{sf} als *die Sprache der Suffixe* von L.

AUFGABE 3.5. Stufe D

Sei $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}$. Wir zeigen nun, dass wenn L regulär ist, dann ist auch L_{sf} regulär.

(a) Geben Sie die Sprache der Suffixe für $\mathsf{L} = \{\mathsf{abc}, \mathsf{d}\}$ an.

(b) Sei L regulär. Beweisen Sie, dass auch L_{sf} regulär ist, indem Sie aus einem NFA $\mathsf{N} = (\mathsf{Q}, \Sigma, \delta, \mathsf{q}_0, \mathsf{F})$ mit $\mathsf{L} = \mathsf{L}(\mathsf{N})$ einen ε-NFA N' mit $\mathsf{L}_{\mathsf{sf}} = \mathsf{L}(\mathsf{N}')$ angeben.

(c) Geben Sie direkt, d.h. ohne den Umweg über NFAs einen regulären Ausdruck r mit $\mathsf{L}(r) = \mathsf{L}((\mathsf{ab} \mid \mathsf{b})^*\mathsf{cd})_{\mathsf{sf}}$ an.

(d) Beschreiben Sie eine rekursive Prozedur, die einen gegebenen regulären Ausdruck r direkt in einen regulären Ausdruck r' mit $\mathsf{L}(r') = \mathsf{L}(r)_{\mathsf{sf}}$ umschreibt.

Erweiterte Lösungsskizze

Seien $\mathsf{L} \subseteq \Sigma^*$ eine beliebige Sprache, $\mathsf{L}_{\mathsf{sf}} := \{v \in \Sigma^* \mid \underbrace{\exists u \in \Sigma^*. uv \in \mathsf{L}}_{\text{Wörter aus L, bei denen vorne ein beliebiger Präfix (auch } \varepsilon \text{ oder das ganze Wort möglich!) abgeschnitten wurde}}\}$ Suffix-Sprache von L.

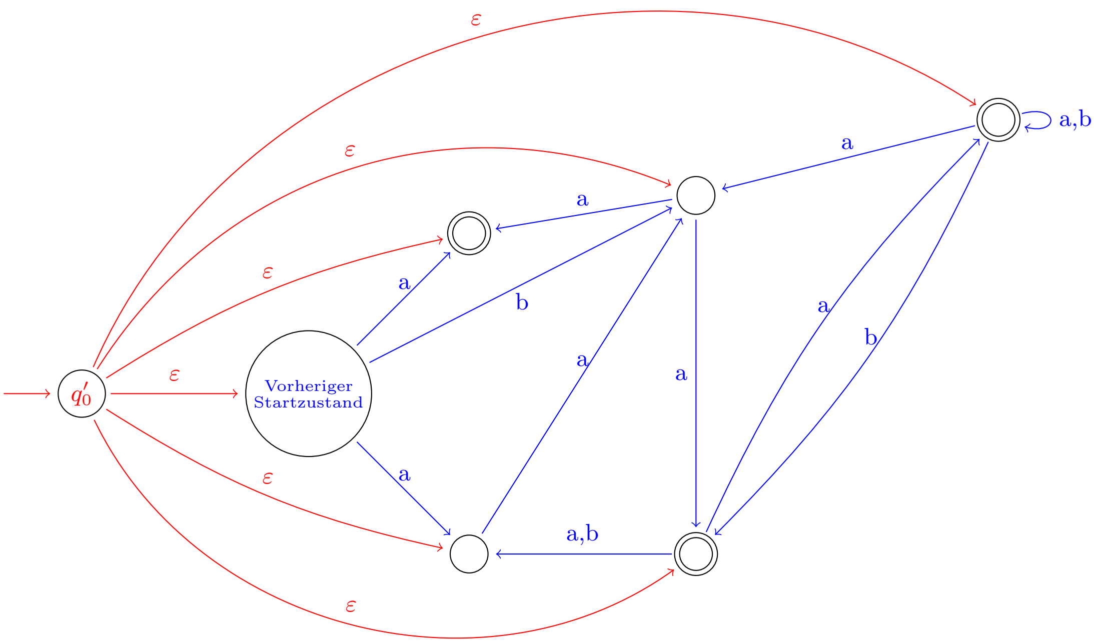
(a) Gib die Suffix-Sprache von $\mathsf{L} = \{\mathsf{abc}, \mathsf{d}\}$ an.

$$\mathsf{L}_{\mathsf{sf}} = \{\underbrace{\mathsf{abc}, \mathsf{bc}, \mathsf{c}, \overbrace{\varepsilon}^{\text{Suffixe von 'd'}}}_{\text{Suffixe von 'abc'}}, \mathsf{d}\}$$

(b) Sei L regulär und $\mathsf{N} = (\mathsf{Q}, \Sigma, \delta, \mathsf{I}, \mathsf{F})$ NFA mit $\mathsf{L}(\mathsf{N}) = \mathsf{L}$. (O.B.d.A. kann vorausgesetzt werden, dass alle Zustände in Q von q_0 aus erreichbar sind: siehe VL) Konstruiere einen ε-NFA N' mit $\mathsf{L}(\mathsf{N}') = \mathsf{L}_{\mathsf{sf}}$.

Idee: Füge einen neuen Startzustand q_0' hinzu, von dem aus man zu jedem Zustand in N springen kann. D.h. man überspringt für ein gegebenes Wort in dessen akzeptierenden Lauf in N ein beliebiges Stück.

Beispiel:



Formale Definition von N':

$$\mathsf{N}' = (\underbrace{\mathsf{Q} \cup \{\mathsf{q}_0'\}}_{\text{Der bisherigen Zustandsmenge Q wird ein neuer Zustand } \mathsf{q}_0' \text{ hinzugefügt}}, \Sigma, \delta', \underbrace{\mathsf{q}_0'}_{\mathsf{q}_0' \text{ wird neuer Start-zustand}}, \mathsf{F}) \quad \text{wobei}$$

$$\delta' = \underbrace{\delta}_{\text{Die alten Transitionen bleiben gleich}} \cup \underbrace{\{(\mathsf{q}_0', \varepsilon, q) | q \in \mathsf{Q}\}}_{\varepsilon\text{-Transitionen von } \mathsf{q}_0' \text{ in jeden anderen Zustand}}$$

Korrektheitsbeweis: Zu zeigen ist die Mengengleichheit $\mathsf{L}(\mathsf{N}') = \mathsf{L}_{\mathsf{sf}}$

Um das zu beweisen, muss gezeigt werden,
dass jedes Wort $w \in \mathsf{L}(\mathsf{N}')$ auch in L_{sf} liegt und
dass jedes Wort $w \in \mathsf{L}_{\mathsf{sf}}$ auch in $\mathsf{L}(\mathsf{N}')$ liegt.

I: $\mathsf{L}_{\mathsf{sf}} \subseteq \mathsf{L}(\mathsf{N}')$: (wenn $v \in \mathsf{L}_{\mathsf{sf}}$, dann auch $v \in \mathsf{L}(\mathsf{N}')$)

Sei $v \in \mathsf{L}_{\mathsf{sf}}$

$\underset{\text{Def. } \mathsf{L}_{\mathsf{sf}}}{\Longrightarrow}$ $\exists u \in \Sigma^*$, sodass $uv \in \mathsf{L}$

$\underset{\text{Def. } N}{\Longrightarrow}$ N akzeptiert das Wort uv für irgendein $u \in \Sigma^*$.

$\underset{\text{Akzeptanzbed. von NFAs}}{\Longrightarrow}$ Es existiert ein akzeptierender Lauf

$$q_0 \xrightarrow{u_1} q_1 \xrightarrow{u_2} \dots \xrightarrow{u_m} q_m \xrightarrow{v_1} q_{m+1} \xrightarrow{v_2} \dots \xrightarrow{v_n} q_{m+n} \in \mathsf{F}$$

des Wortes uv im NFA N'.

$\underset{\text{Def. } \mathsf{N}'}{\Longrightarrow}$ Es existiert eine Transition $q_0' \xrightarrow{\varepsilon} q_m$
im konstruierten ε-NFA N' .
Dann existiert aber auch der Pfad

$\Longrightarrow$ Der ε-NFA N' wurde genau so definiert, dass vom neuen Startzustand q_0' aus jeder Zustand über eine ε-Transition erreichbar ist.

$$q_0' \xrightarrow{\varepsilon} q_m \xrightarrow{v_1} q_{m+1} \xrightarrow{v_2} \dots \xrightarrow{v_n} q_{m+n} \in \mathsf{F}' = \mathsf{F}$$

im konstruierten Automaten.
Dieser ist (offensichtlich) akzeptierender Lauf des Wortes v .

$\underset{\text{Akzeptanzbed. von } \varepsilon\text{-NFAs}}{\Longrightarrow}$ $v \in \mathsf{L}(\mathsf{N}')$

II: $\mathsf{L}(\mathsf{N}') \subseteq \mathsf{L}_{\mathsf{sf}}$: (wenn $v \in \mathsf{L}(\mathsf{N}')$, dann auch $v \in \mathsf{L}_{\mathsf{sf}}$)

Sei $v \in \mathsf{L}(\mathsf{N}')$

$\underset{\text{Akzeptanzbed. von } \varepsilon\text{-NFAs}}{\Longrightarrow}$ Es existiert ein akzeptierender Lauf
des Wortes v im NFA N':

$$q_0' \xrightarrow{\varepsilon} q_1 \xrightarrow{v_1} q_2 \xrightarrow{v_2} \dots \xrightarrow{v_n} q_n \in F' = F$$

$\Longrightarrow$ Jeder Lauf eines Wortes in N' beginnt mit einer ε-Transition. Befindet man sich in einem Zustand $q \neq q_0'$, so kann man keine ε-Transition nehmen, muss also ein Zeichen von v einlesen.

$\underset{\text{Jeder Zustand in } N \text{ ist erreichbar}}{\Longrightarrow}$ In N existiert ein Pfad:

$$q_0 \xrightarrow{u_1} q_x \xrightarrow{u_2} \dots \xrightarrow{u_m} q_1$$

da q_1 in N von q_0 aus erreichbar ist.
Über diesen Pfad wird irgendein Wort $u = u_1 u_2 \dots u_m \in \Sigma^*$
eingelesen.

Damit existiert aber auch der Pfad:

$$q_0 \xrightarrow{u_1} q_x \xrightarrow{u_2} \dots \xrightarrow{u_m} q_1 \xrightarrow{v_1} q_2 \xrightarrow{v_2} \dots \xrightarrow{v_n} q_n \in F' = F$$

$\underset{\text{Akzeptanzbed. von NFAs}}{\Longrightarrow}$ N akzeptiert das Wort uv für irgendein $u \in \Sigma^*$.

$\Longrightarrow$ $\exists u \in \Sigma^*$, sodass $uv \in \mathsf{L}$

$\underset{\text{Def. } \mathsf{L}_{\mathsf{sf}}}{\Longrightarrow}$ $v \in \mathsf{L}_{\mathsf{sf}}$

AUFGABE 3.6.

Stufe D

Sei Σ ein endliches Alphabet. Ein *sternfreier* Ausdruck ist wie folgt definiert:

- *Syntax:* Die Grammatik mit folgender Produktion gibt die Menge aller gültigen sternfreien Ausdrücke an:

$$\mathsf{S} \to \boldsymbol{\emptyset} \mid \boldsymbol{\varepsilon} \mid x \mid \mathsf{SS} \mid \overline{\mathsf{S}} \mid \mathsf{S|S} \qquad\qquad x \in \Sigma$$

  Wir bezeichnen mit | Vereinigung, mit SS Konkatenation und mit $\overline{\mathsf{S}}$ Komplement.

- *Semantik*

$$\mathsf{L}(\boldsymbol{\emptyset}) := \emptyset \quad \mathsf{L}(\boldsymbol{\varepsilon}) := \{\varepsilon\} \quad \mathsf{L}(\mathsf{a}) := \{\mathsf{a}\} \quad \mathsf{L}(\alpha\beta) := \mathsf{L}(\alpha)\mathsf{L}(\beta) \quad \mathsf{L}(\overline{\alpha}) := \Sigma^* \setminus \mathsf{L}(\alpha) \quad \mathsf{L}(\alpha|\beta) := \mathsf{L}(\alpha) \cup \mathsf{L}(\beta)$$

(a) Zeigen Sie mittels struktureller Induktion, dass $\mathsf{L}(\alpha)$ eine reguläre Sprache über Σ ist, falls α ein sternfreier Ausdruck über Σ ist. Passen Sie hierfür den Beweis aus den Folien, dass jeder reguläre Ausdruck sich in einen NFA übersetzen lässt, entsprechend an.

Hinweise:
- Die Umkehrung von (a) gilt nicht, z.B. ist $\mathsf{L}((\mathsf{aa})^*)$ eine reguläre Sprache über $\Sigma = \{\mathsf{a}\}$, welche sich nicht durch einen sternfreien Ausdruck beschreiben lässt.
- Sie dürfen ohne Beweis die folgende Aussage verwenden: Ist $\mathsf{D} = (\mathsf{Q}, \Sigma, \delta, \mathsf{q}_0, \mathsf{F})$ ein DFA, so akzeptiert der DFA $\mathsf{D}' = (\mathsf{Q}', \Sigma', \delta', \mathsf{q}_0', \mathsf{F}')$, wobei $\mathsf{Q} = \mathsf{Q}'$, $\Sigma = \Sigma'$, $\mathsf{q}_0 = \mathsf{q}_0'$, $\delta = \delta'$ und $\mathsf{F}' = \mathsf{Q} \setminus \mathsf{F}$ gerade die Sprache $\mathsf{L}(\mathsf{D}') = \Sigma^* \setminus \mathsf{L}(\mathsf{D})$.
- (a) impliziert, dass es für jeden sternfreien Ausdruck α einen regulären Ausdruck α' mit $\mathsf{L}(\alpha) = \mathsf{L}(\alpha')$ gibt.

(b) Beweisen Sie die Behauptung $\mathsf{L}(\boldsymbol{\varepsilon}|\overline{\overline{\mathsf{a}\overline{\boldsymbol{\emptyset}}}|\overline{\overline{\boldsymbol{\emptyset}}\mathsf{b}}|\overline{\boldsymbol{\emptyset}}\mathsf{aa}\overline{\boldsymbol{\emptyset}}|\overline{\boldsymbol{\emptyset}}\mathsf{bb}\overline{\boldsymbol{\emptyset}}}) = \mathsf{L}\big((\mathsf{ab})^*\big)$ für $\Sigma = \{\mathsf{a}, \mathsf{b}\}$, indem Sie
  (i) zuerst den sternfreien Ausdruck in einen $\boldsymbol{\varepsilon}$-NFA übersetzen,
  (ii) den $\boldsymbol{\varepsilon}$-NFA gemäß der Vorlesung über Zwischenschritte in einen regulären Ausdruck übertragen und
  (iii) schließlich den erhaltenen regulären Ausdruck mittels der Äquivalenzen aus der Vorlesung zu $(\mathsf{ab})^*$ vereinfachen.

Hinweis: Vereinfachen Sie den NFA zunächst vor der Übersetzung in einen regulären Ausdruck, indem Sie alle Zustände, die keinen Endzustand erreichen können, entfernen. Erklären Sie kurz, warum sich die erkannte Sprache dadurch nicht ändert.

AUFGABE 3.7.

Stufe E

Wir betrachten das duale Model zu NFAs und definieren für diese Aufgabe *UFA*s (universelle endliche Automaten), die ein Wort w akzeptieren gdw. alle Läufe auf w in einem Endzustand enden. Sei $\mathsf{U} = (\mathsf{Q}, \Sigma, \delta, \mathsf{q}_0, \mathsf{F})$ ein UFA. Dann wird das Wort $w \in \Sigma^*$ genau dann akzeptiert, wenn $\delta(\mathsf{q}_0, w) \subseteq \mathsf{F}$.

(a) Entscheiden Sie, ob die folgende Aussage gilt und begründen Sie Ihre Antwort, indem Sie einen passenden Beweis oder ein passendes Gegenbeispiel angeben:

  Jeder UFA ohne Endzustände akzeptiert die leere Sprache.

(b) Wir betrachten folgende Sprache $\mathsf{L}_k = \{w \in \Sigma^* \mid \forall 1 \leq i \leq |w|\,.\; w_i = \mathsf{a} \to w_{i+k} = \mathsf{b}\}$ über dem Alphabet $\Sigma = \{\mathsf{a}, \mathsf{b}\}$. Konstruieren Sie einen NFA und einen UFA für $k = 2$ und $k = 3$. Vergleichen Sie Ihre beiden Lösung bezüglich der Größe der Automaten.

(c) Geben Sie eine Übersetzung von UFAs zu DFAs an und beweisen Sie deren Korrektheit.