2. Übungsblatt (mit Lösungen)

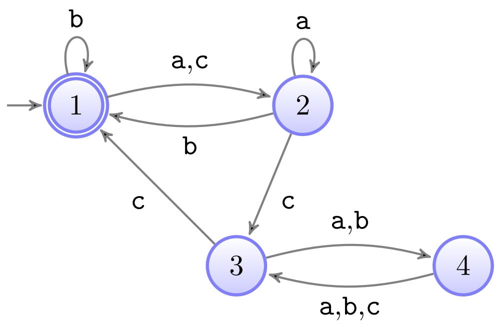
3.0 VU Formale Modellierung

Marion Scholz, Gernot Salzer

18. April 2019

Aufgabe 1 (3 Punkte)

Sei $\mathcal{A}$ der folgende endliche Automat.



(a) Geben Sie alle Wörter an, die aus maximal drei Zeichen bestehen und von $\mathcal{A}$ akzeptiert werden. (Das sollten 14 Wörter sein.)

(b) Berechnen Sie schrittweise $\delta^*(1, \texttt{bacabc})$.

(c) Spezifizieren Sie $\mathcal{A}$ in tabellarischer Form. Handelt es sich bei $\mathcal{A}$ um einen deterministischen oder indeterministischen Automaten?

Lösung

(a) Die von $\mathcal{A}$ akzeptierten Wörter mit einer Länge von maximal drei Zeichen sind ε, `b`, `ab`, `bb`, `cb`, `aab`, `abb`, `acc`, `bab`, `bbb`, `bcb`, `cab`, `cbb` und `ccc`.

(b) $\delta^*(1, \texttt{bacabc}) = \delta^*(\delta(1, \texttt{b}), \texttt{acabc})$
$= \delta^*(1, \texttt{acabc})$
$= \delta^*(\delta(1, \texttt{a}), \texttt{cabc})$
$= \delta^*(2, \texttt{cabc})$
$= \delta^*(\delta(2, \texttt{c}), \texttt{abc})$
$= \delta^*(3, \texttt{abc})$
$= \delta^*(\delta(3, \texttt{a}), \texttt{bc})$
$= \delta^*(4, \texttt{bc})$
$= \delta^*(\delta(4, \texttt{b}), \texttt{c})$
$= \delta^*(3, \texttt{c})$
$= \delta^*(\delta(3, \texttt{c}), \varepsilon)$
$= \delta^*(1, \varepsilon)$
$= 1$

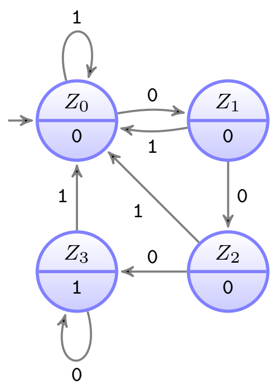
(c) $\mathcal{A} = \langle\{1, 2, 3, 4\},\ \{\texttt{a}, \texttt{b}, \texttt{c}\},\ \delta,\ 1,\ \{1\}\rangle$, wobei die Übergangsfunktion δ durch folgende Tabelle definiert ist:

δ	a	b	c
1	2	1	2
2	2	1	3
3	4	4	1
4	3	3	3

$\mathcal{A}$ ist ein deterministischer Automat, da der momentane Zustand und die nächste Eingabe immer eindeutig den Folgezustand bestimmen. Das äußert sich in der Tabelle dadurch, dass jeder Eintrag genau einen Zustand enthält.

Aufgabe 2 **(3 Punkte)**

Sei $\mathcal{A}$ der folgende Moore-Automat.



(a) Geben Sie die Ausgaben zu folgenden Eingaben an: 10101, 10001, 00000.

(b) Berechnen Sie schrittweise $\delta^*(Z_0, \texttt{00001})$ und $\gamma^*(Z_0, \texttt{00001})$.

(c) Beschreiben Sie die Übersetzungsfunktion $[\mathcal{A}]$.

Lösung

(a)

w:	10101	10001	00000
$[\mathcal{A}](w)$:	000000	000010	000111

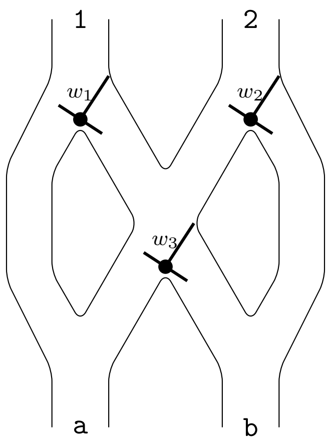
(b)
$$\begin{aligned}
\delta^*(Z_0, \texttt{00001}) &= \delta^*(\delta(Z_0, \texttt{0}), \texttt{0001}) = \delta^*(Z_1, \texttt{0001}) \\
&= \delta^*(\delta(Z_1, \texttt{0}), \texttt{001}) \quad = \delta^*(Z_2, \texttt{001}) \\
&= \delta^*(\delta(Z_2, \texttt{0}), \texttt{01}) \quad\;\; = \delta^*(Z_3, \texttt{01}) \\
&= \delta^*(\delta(Z_3, \texttt{0}), \texttt{1}) \quad\;\;\; = \delta^*(Z_3, \texttt{1}) \\
&= \delta^*(\delta(Z_3, \texttt{1}), \varepsilon) \quad\;\;\; = \delta^*(Z_0, \varepsilon) \\
&= Z_0
\end{aligned}$$

$$\begin{aligned}
\gamma^*(Z_0, \texttt{00001}) &= \gamma(Z_0) \cdot \gamma^*(\delta(Z_0, \texttt{0}), \texttt{0001}) &&= \texttt{0} \cdot \gamma^*(Z_1, \texttt{0001}) \\
&= \texttt{0} \cdot \gamma(Z_1) \cdot \gamma^*(\delta(Z_1, \texttt{0}), \texttt{001}) &&= \texttt{00} \cdot \gamma^*(Z_2, \texttt{001}) \\
&= \texttt{00} \cdot \gamma(Z_2) \cdot \gamma^*(\delta(Z_2, \texttt{0}), \texttt{01}) &&= \texttt{000} \cdot \gamma^*(Z_3, \texttt{01}) \\
&= \texttt{000} \cdot \gamma(Z_3) \cdot \gamma^*(\delta(Z_3, \texttt{0}), \texttt{1}) &&= \texttt{0001} \cdot \gamma^*(Z_3, \texttt{1}) \\
&= \texttt{0001} \cdot \gamma(Z_3) \cdot \gamma^*(\delta(Z_3, \texttt{1}), \varepsilon) &&= \texttt{00011} \cdot \gamma^*(Z_0, \varepsilon) \\
&= \texttt{00011} \cdot \gamma(Z_0) &&= \texttt{000110}
\end{aligned}$$

(c) Der Automat erkennt drei oder mehr aufeinanderfolgende `0`er in der Eingabe: Werden mindestens drei `0`er eingelesen, so ist die Ausgabe `1`, sonst `0`.

Aufgabe 3 (3 Punkte)

Max erhält zu Weihnachten ein Knobelspiel, das aus einem Würfel und einer Murmel (Glaskugel) besteht. Der Würfel besitzt oben und unten jeweils zwei Löcher. Wirft man die Murmel bei einem der beiden oberen Löcher hinein, kommt sie bei einem der beiden unteren wieder heraus. Die Rätsel besteht nun darin vorherzusagen, bei *welchem* der beiden Löcher sie unten auftauchen wird.
Am Ende der Weihnachtsferien verliert Max schließlich die Geduld und zerlegt den Würfel. Er macht folgende Skizze vom Inneren des Würfels.



Die Löcher oben bezeichnet er mit `1` und `2`, die beiden unten mit `a` und `b`. Dort, wo sich die Murmelbahnen gabeln, befinden sich Weichen (w_1 bis w_3), die eine Bahn offen lassen und die andere verschließen. Kommt die Murmel an diese Stelle, kann sie nur die offene Bahn nehmen. Sobald die Murmel an der Weiche vorbei ist, schaltet sie durch einen kleinen Hebel die Weiche hinter sich um, sodass die Murmel das nächste Mal an dieser Stelle die andere Abzweigung nehmen muss. Bei der Weichenstellung laut Skizze wird die Murmel bei jeder Weiche nach links geleitet, sodass sie von beiden Eingängen, `1` und `2`, zum Ausgang `a` gelangt. Beim Weg von `1` nach `a` wird dabei Weiche w_1 umgeschaltet, beim Weg von `2` nach `a` werden hingegen die Weichen w_2 und w_3 beide umgeschaltet.
Die Reihenfolge, in der die Murmel in die beiden Löcher oben geworfen wird, lässt sich durch ein Wort über $\{\texttt{1}, \texttt{2}\}$ beschreiben. Etwa bedeutet `121`, dass die Murmel zuerst in das Loch `1`, dann in das Loch `2` und zuletzt wieder in das Loch `1` geworfen wird. Analog lässt sich die Reihenfolge der Löcher, bei denen die Murmel unten herauskommt, durch ein Wort über $\{\texttt{a}, \texttt{b}\}$ beschreiben. Stehen die Weichen zu Beginn so wie in der Skizze gezeigt, dann führt das Eingabewort `121` zur Ausgabe `aab`.
Modellieren Sie das Verhalten dieses Spiels mit Hilfe eines Mealy-Automaten, der `1`/`2`-in `a`/`b`-Wörter umwandelt. Verwenden Sie die skizzierte Weichenstellung als Anfangszustand. Sie können die Übergangs- und Ausgabefunktion graphisch oder als Tabelle darstellen.
Berechnen Sie weiters $\gamma^*(q_0, \texttt{1121122})$, wobei γ die Ausgabefunktion und q_0 den Anfangszustand Ihres Automaten bezeichnet.[1]

Lösung

Der Zustand des Würfels wird durch die Stellung der drei Weichen festgelegt. Wir bezeichnen die Zustände mit $w_1w_2w_3$, wobei $w_i = 0$ bedeutet, dass die Weiche rechts steht (wie in der Skizze eingezeichnet), und $w_i = 1$, dass sich die Weiche in der linken Stellung befindet. Das Spiel wird durch den Mealy-Automaten

$$\langle \{000, 001, \ldots, 111\}, \{\texttt{1}, \texttt{2}\}, \{\texttt{a}, \texttt{b}\}, \delta, \gamma, 000 \rangle$$

beschrieben, wobei die Übergangsfunktion δ und die Ausgabefunktion γ durch folgende Tabelle definiert sind.

	δ		γ	
	`1`	`2`	`1`	`2`
000	100	011	`a`	`a`
001	101	010	`a`	`b`
010	110	000	`a`	`b`
011	111	001	`a`	`b`
100	001	111	`a`	`a`
101	000	110	`b`	`b`
110	011	100	`a`	`b`
111	010	101	`b`	`b`

Für die Eingabe `1121122` erhalten wir $\gamma^*(000, \texttt{1121122}) = \texttt{aabaabb}$.

[1] Inspiriert von einer Aufgabe unter `bytesoftheday.wordpress.com/category/theory-of-computation/`

Aufgabe 4 (3 Punkte)

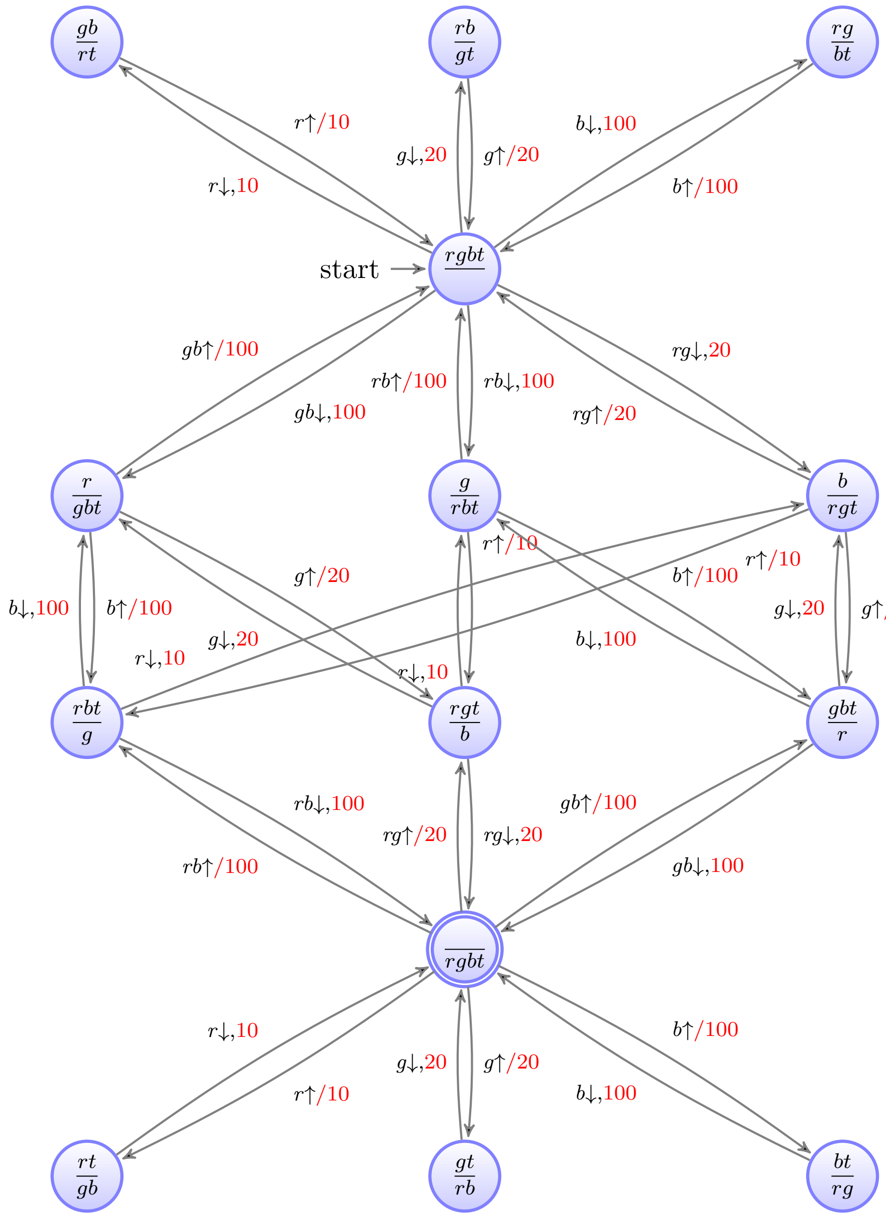
Drei Zauberer wollen nach Hause zu ihrer Höhle, der Weg wird ihnen aber von einem dichten violetten Nebel versperrt. Mithilfe einer Zauberlaterne können sie im Nebel sehen und den Weg finden. Die Laterne leuchtet aber nur hell genug, um zwei Zauberern gleichzeitig den Weg zur Höhle zeigen zu können. Die drei Zauberer sind unterschiedlich gut zu Fuß. Der rote Zauberer benötigt für die Strecke 10 Minuten und der gelbe 20 Minuten. Der blaue Zauberer hat sich den Knöchel verstaucht und braucht sogar eine Stunde und 40 Minuten.[2]

- Überlegen Sie, welche Informationen notwendig sind, um den Zustand des Systems zu beschreiben.
- Legen Sie die möglichen Aktionen fest, die zu einem Zustandswechsel führen können.
- Geben Sie einen endlichen Automaten an, der das Systemverhalten vollständig beschreibt. Aus dem Automaten sollen sich alle Möglichkeiten ablesen lassen, wie die Zauberer nach Hause kommen können.
- Fügen Sie bei den einzelnen Übergängen die Zeit hinzu, die die jeweilige Aktion benötigt. Welche der im letzten Punkt gefundenen Möglichkeiten benötigen am wenigsten Zeit, das heißt, welche Möglichkeiten bringen die Zauberer am schnellsten nach Hause?

Lösung

Der momentane Zustand des Systems wird durch die Information beschrieben, welche der Zauberer sich noch vor dem Nebel befinden und welche schon zu Hause sind und wo sich die Taschenlampe befindet. Wenn man mit r, g, b und t den roten, gelben, blauen Zauberer bzw. die Taschenlampe symbolisiert, den Nebel mit einem Querstrich, oben mit „vor dem Nebel“ und unten mit „zu Hause“, dann kann man die Zustände kompakt durch $\frac{abct}{}$ (Anfangszustand), $\frac{}{abct}$ (Endzustand), $\frac{a}{bct}$, … beschreiben.
Die Aktionen bestehen darin, dass einer oder zwei der Zauberer mit der Taschenlampe den Nebel durchqueren. Da die Taschenlampe ohnehin immer dabei sein muss, bezeichnen wir die Aktionen nur mit den Kürzeln der Zauberer, die unterwegs sind. Mit einem Pfeil geben wir die Richtung an: $\downarrow$ bedeutet „nach Hause“, $\uparrow$ bedeutet „von zu Hause weg“. Die Bezeichung $rg\downarrow$ bedeutet also, dass der rote und der gelbe Zauberer nach Hause gehen, während $b\uparrow$ bedeutet, dass der blaue Zauberer von zu Hause wieder zurück geht. Die möglichen Aktionen sind also $r\downarrow$, $g\downarrow$, $b\downarrow$, $rg\downarrow$, $rb\downarrow$, $gb\downarrow$, $r\uparrow$, $g\uparrow$, $b\uparrow$, $rg\uparrow$, $rb\uparrow$ und $gb\uparrow$.

[2] In Anlehnung an ein Level des Spiels „I Have 1 Day“ von Max Games



Die schnellsten Lösungen sind jene, bei denen der rote Zauberer nochmals zurückgeht, um den zweiten Kollegen zu holen. Es gibt zwei davon, die jeweils 130 Minuten benötigen:

- $rb\!\downarrow\ r\!\uparrow\ rg\!\downarrow$, $100 + 10 + 20 = 130\,\text{min}$
- $rg\!\downarrow\ r\!\uparrow\ rb\!\downarrow$, $20 + 10 + 100 = 130\,\text{min}$

Aufgabe 5 (4 Punkte)

Sei Σ das Alphabet $\{\texttt{a}, \texttt{h}, \texttt{n}, \texttt{s}\}$ und L die Menge aller Wörter über Σ, die entweder mit `hans` oder `anna` enden. Beispiele für solche Wörter sind `hans` und `anna` selber, aber auch die Wörter `hahahans` und `hanna` liegen in L.

(a) Geben Sie einen Posix Extended Regular Expression an, der die Sprache L beschreibt.

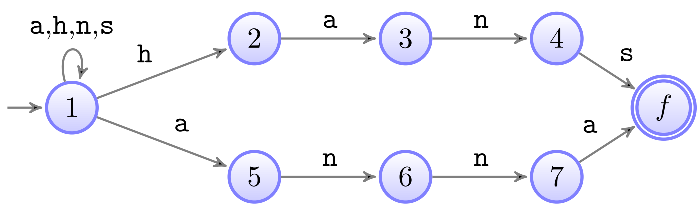
(b) Geben Sie einen nichtdeterministischen Automaten an, der die Sprache L akzeptiert. Der Automat soll der Definition der Sprache direkt entsprechen, sodass die Korrektheit der Modellierung unmittelbar einsichtig ist.

(c) Konstruieren Sie mit Hilfe des in der Vorlesung besprochenen Determinisierungsverfahrens zu Ihrem nichtdeterministischen Automaten einen äquivalenten deterministischen.

Lösung

(a) `[ahns]*(hans|anna)` oder `^[ahns]*(hans|anna)$` (falls die Zeichenkette die gesamte Zeile einnehmen soll)

(b) Ein indeterministischer Automat, der diese Sprache darstellt, ist der folgende:



(c) Wir stellen die Übergangsfunktion als Tabelle dar, da diese besser als Ausgangsbasis für die Determinisierung geeignet ist. (Genauer: Wir bestimmen das Ergebnis der erweiterten Übergangsfunktion δ^* für jeden Zustand und jedes Eingabesymbol. Wenn es ε-Übergänge gibt, müssen auch längere Pfade betrachtet werden.)

δ^*	a	h	n	s
1	$\{1,5\}$	$\{1,2\}$	$\{1\}$	$\{1\}$
2	$\{3\}$	$\{\}$	$\{\}$	$\{\}$
3	$\{\}$	$\{\}$	$\{4\}$	$\{\}$
4	$\{\}$	$\{\}$	$\{\}$	$\{f\}$
5	$\{\}$	$\{\}$	$\{6\}$	$\{\}$
6	$\{\}$	$\{\}$	$\{7\}$	$\{\}$
7	$\{f\}$	$\{\}$	$\{\}$	$\{\}$
f	$\{\}$	$\{\}$	$\{\}$	$\{\}$

Einen deterministischen Automaten erhalten wir, indem wir den indeterminischen Automaten simulieren. Ein Zustand des deterministischen Automaten repräsentiert dabei jene Zustände des indeterministischen, in denen sich dieser zu diesem Zeitpunkt befinden kann. Der Startzustand wird mit $\{1\}$ bezeichnet, da sich der indeterministische Automat zu Beginn im Zustand 1 (und nur in diesem) befindet. Von diesem Zustand ausgehend erstellen wir zeilenweise die Tabelle für die Übergangsfunktion des deterministischen Automaten.

$\hat{\delta}$	a	h	n	s
$\{1\}$	$\{1,5\}$	$\{1,2\}$	$\{1\}$	$\{1\}$
$\{1,2\}$	$\{1,3,5\}$	$\{1,2\}$	$\{1\}$	$\{1\}$
$\{1,5\}$	$\{1,5\}$	$\{1,2\}$	$\{1,6\}$	$\{1\}$
$\{1,6\}$	$\{1,5\}$	$\{1,2\}$	$\{1,7\}$	$\{1\}$
$\{1,7\}$	$\{1,5,f\}$	$\{1,2\}$	$\{1\}$	$\{1\}$
$\{1,3,5\}$	$\{1,5\}$	$\{1,2\}$	$\{1,4,6\}$	$\{1\}$
$\{1,4,6\}$	$\{1,5\}$	$\{1,2\}$	$\{1,7\}$	$\{1,f\}$
$\{1,f\}$	$\{1,5\}$	$\{1,2\}$	$\{1\}$	$\{1\}$
$\{1,5,f\}$	$\{1,5\}$	$\{1,2\}$	$\{1,6\}$	$\{1\}$

Jene Zustände, die einer Situation entsprechen, in der der indeterministische Automat einen Endzustand erreicht hat, sind die Endzustände des deterministischen Automaten; in diesem Beispiel sind das alle Zustände, deren Bezeichnung f enthält. Dieser wird somit durch das Tupel $\langle \widehat{Q}, \Sigma, \hat{\delta}, \{1\}, \widehat{F} \rangle$ beschrieben, wobei

$$\begin{aligned}
\Sigma &= \{\mathtt{a}, \mathtt{h}, \mathtt{n}, \mathtt{s}\} \\
\widehat{F} &= \{\{1,f\}, \{1,5,f\}\} \\
\widehat{Q} &= \{\{1\}, \{1,2\}, \{1,5\}, \{1,6\}, \{1,7\}, \{1,3,5\}, \{1,4,6\}\} \cup \widehat{F}
\end{aligned}$$

Aufgabe 6 (2 Punkte)

Geben Sie endliche Automaten an, die dieselben Sprachen beschreiben wie die folgenden regulären Ausdrücke in algebraischer Notation.

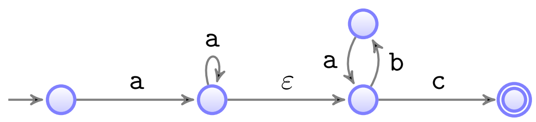
(a) $\mathtt{aa}^*(\mathtt{ba})^*\mathtt{c}$

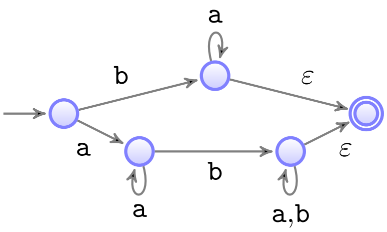
(b) $\mathtt{ba}^* + \mathtt{aa}^*\mathtt{b}(\mathtt{a}+\mathtt{b})^*$

Lösung

Die gesuchten Automaten können mit dem allgemeinen Verfahren konstruiert werden, enthalten dann aber in der Regel viel mehr Zustände und ε-Kanten als notwendig. Die folgenden Automaten wurden bereits vereinfacht.

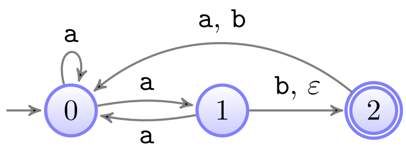
(a)



(b) 

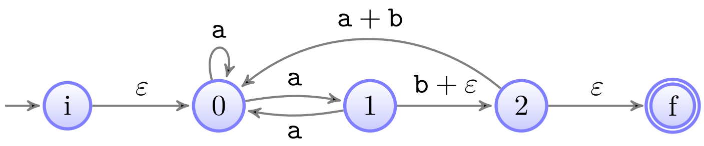
Aufgabe 7 (4 Punkte)

Konstruieren Sie zu folgendem endlichen Automaten einen regulären Ausdruck. Orientieren Sie sich am Algorithmus, der in der Vorlesung besprochen wurde und geben Sie den Automaten nach jeder Zustandselimination an!



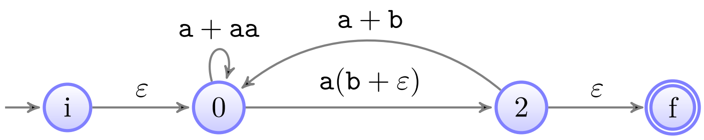
Lösung

Neuer Anfangs- und Endzustand:

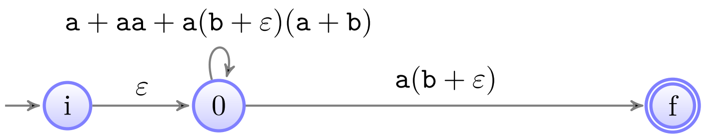


Wir eliminieren die Zustände in der Reihenfolge 1, 2 und 0; die anderen Reihenfolgen sind ebenfalls möglich.

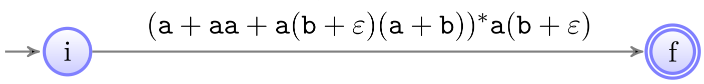
Elimination von Zustand 1:



Elimination von Zustand 2:



Elimination von Zustand 0:



Die Sprache des ursprünglichen Automaten wird also durch den Ausdruck

$$(\mathtt{a}+\mathtt{aa}+\mathtt{a}(\mathtt{b}+\varepsilon)(\mathtt{a}+\mathtt{b}))^*\mathtt{a}(\mathtt{b}+\varepsilon)$$

beschrieben.

Aufgabe 8 (3 Punkte)

Fahrgestellnummern von Opel-PKWs ab Modelljahr 1998 sind wie folgt aufgebaut[3]:

- 1.- 3./4. Stelle: Weltherstellercode; 3-4 Zeichen bestehend aus Großbuchstaben und der Ziffer 0
- 4. Stelle[4]: Fahrzeugtyp; „0“ = kein Sonderfahrzeug bzw. „P“ = Postfahrzeug
- 5. Stelle: GM-Code Plattform; ein Großbuchstabe
- 6. Stelle: GM-Code Modellbezeichnung; einer der Großbuchstaben A, B, C, D, E, F, G, H
- 7. Stelle: GM-Code Karosserieform; ein Großbuchstabe
- 8. + 9. Stelle: GM-Code Karosserieform zweistellige Zahl von 01 bis 99
- 10. Stelle: Modelljahr; ein Großbuchstabe oder eine Ziffer 1-9
- 11. Stelle: Herstellerwerk; ein Großbuchstabe oder eine Ziffer 0-9
- ab 12. Stelle: Fortlaufende Fahrzeugnummer; beliebig lange Nummer mit mindestens einer Ziffer

W0L0EAR971N0123456 ist beispielsweise die Fahrgestellnummer von einem konkreten Opel Speedster.
Beschreiben Sie den Aufbau solcher Fahrgestellnummern mit den folgenden Methoden. Treffen Sie sinnvolle Annahmen, wenn Ihnen Informationen fehlen.

(a) Geben Sie einen regulären Ausdruck in algebraischer Notation an.

(b) Geben Sie einen regulären Ausdruck in Posix-Notation an, der alle Zeilen beschreibt, die *ausschließlich* eine derartige Fahrgestellnummer enthalten.

(c) Zeichnen Sie das Syntaxdiagramm, das Ihrem regulären Ausdruck aus Teil a entspricht.

[3] `https://www.opel-infos.de/fgst/fgst9_97.html`
[4] bzw. 5. Stelle falls der Weltherstellercode 4-stellig ist

Lösung

(a) Fahrgestellnummern können durch den Ausdruck

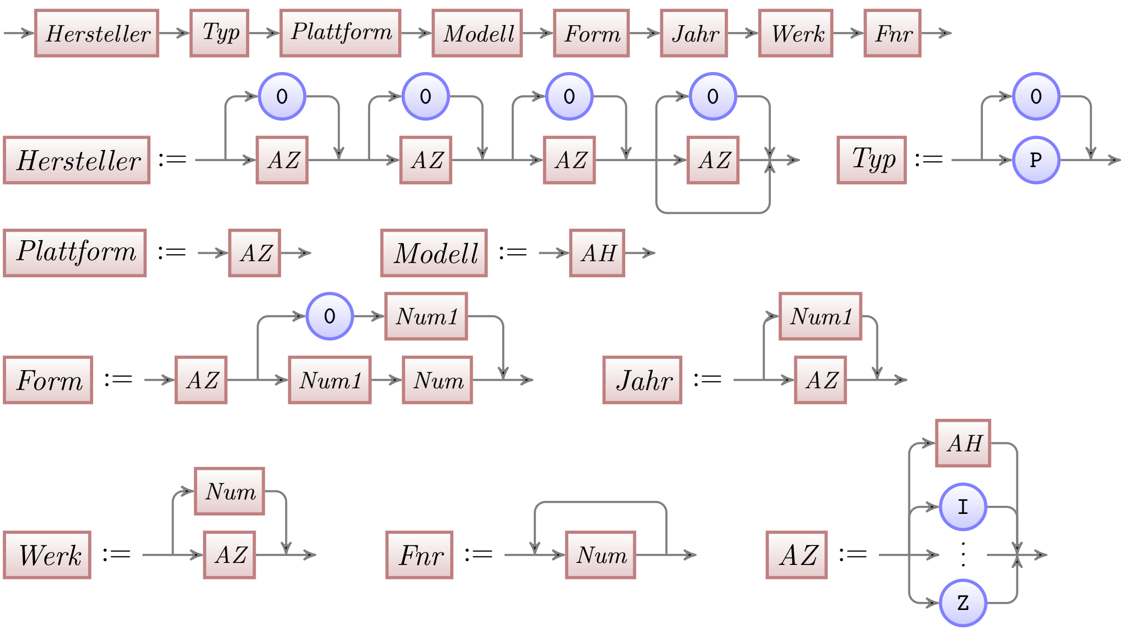
$$\textit{Hersteller Typ Plattform Modell Form Jahr Werk Fnr}$$

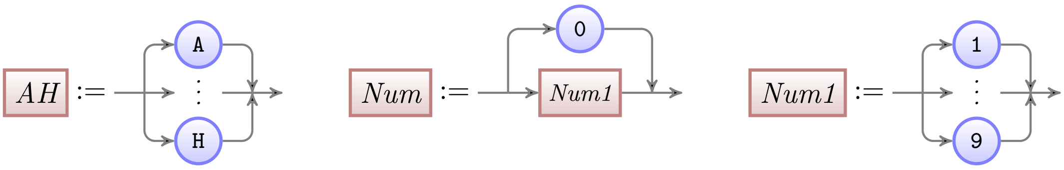
beschrieben werden, wobei wir folgende Abkürzungen verwenden:

$$\begin{aligned}
\textit{Hersteller} &:= (AZ + \mathtt{0})(AZ + \mathtt{0})(AZ + \mathtt{0})(AZ + \mathtt{0} + \varepsilon) \\
\textit{Typ} &:= (\mathtt{0} + \mathtt{P}) \\
\textit{Plattform} &:= AZ \\
\textit{Modell} &:= AH \\
\textit{Form} &:= AZ\,(\mathtt{0}\,\textit{Num1} + \textit{Num1}\ \textit{Num}) \\
\textit{Jahr} &:= (AZ + \textit{Num1}) \\
\textit{Werk} &:= (AZ + \textit{Num}) \\
\textit{Fnr} &:= \textit{Num}\ \textit{Num}^* \\
AZ &:= (AH + \mathtt{I} + \cdots + \mathtt{Z}) \\
AH &:= (\mathtt{A} + \cdots + \mathtt{H}) \\
\textit{Num} &:= (\mathtt{0} + \textit{Num1}) \\
\textit{Num1} &:= (\mathtt{1} + \cdots + \mathtt{9})
\end{aligned}$$

(b) `^[A-Z0]{3,4}[0P][A-Z][A-H][A-Z](0[1-9]|[1-9][0-9])[A-Z1-9][A-Z0-9][0-9]+$`

(c) Syntaxdiagramm:





Aufgabe 9 (5 Punkte)

Mit den drei Zeichen `/`, `_` und `\` lassen sich Bergsilhouetten[5] skizzieren, wie etwa die folgende:

```
            _
         /\/ \_
      /\/      \
   __/          \_
 _/               \
/                  \___/\_
```

Diese Silhouette lässt sich platzsparender auch in eine einzige Zeile zusammenschieben (komprimieren):

```
/_/__//\//\/_\_\\_\\___/\_
```

Wir legen fest, dass Silhouetten immer auf derselben Grundlinie aufhören, auf der sie begonnen haben, und dass kein Teil der Silhouette unter dieser Grundlinie liegt. Weiters kann es auf jeder Höhe beliebig breite Plateaus (Ebenen) geben. Ob die leere Silhouette bereits eine Silhouette darstellt, bleibt Ihnen überlassen.[6]
Sei $\mathcal{S}$ die Menge aller Zeichenketten, die eine Silhouette in komprimierter Form darstellen.

(a) Beschreiben Sie die Sprache $\mathcal{S}$ mit Hilfe einer kontextfreien Grammatik. Verwenden Sie EBNF-Notationen, um die Grammatik übersichtlich zu strukturieren.

(b) Ist die leere Silhouette laut Ihrer Grammatik eine zulässige Silhouette? Begründen Sie Ihre Antwort.

(c) Zeigen Sie, dass die komprimierte Silhouette `//\/_\_\` in der Sprache Ihrer Grammatik liegt.

(d) Handelt es sich bei $\mathcal{S}$ um eine reguläre Sprache, d.h., lässt sich diese Sprache im Prinzip auch durch einen (komplizierten) regulären Ausdruck spezifizieren? Begründen Sie Ihre Antwort.

[5] *Silhouette* ist ein anderes Wort für *Umriss*.
[6] Inspiriert von einer Aufgabe von M.Schmidt-Schauß, Goethe-Universität Frankfurt.

Lösung

(a) $\mathcal{S} = \mathcal{L}(\langle N, T, P, \mathit{Silhouette} \rangle)$, wobei

$$\begin{aligned}
N &= \{\mathit{Silhouette}, \mathit{Plateau}, \mathit{Berg}\},\\
T &= \{\texttt{"/"}, \texttt{"_"}, \texttt{"\textbackslash"}\},\\
P &= \{\ \mathit{Silhouette} \to \{\ \mathit{Plateau} \mid \mathit{Berg}\ \}\ ,\\
&\qquad \mathit{Plateau} \to \texttt{"_"}\ ,\\
&\qquad\quad \mathit{Berg} \to \texttt{"/"}\ \mathit{Silhouette}\ \texttt{"\textbackslash"}\ \ \}.
\end{aligned}$$

(b) Ja, die leere Silhouette liegt in $\mathcal{S}$, da der Ausdruck $\{\cdots\}$ auf der rechten Seite der Produktion für *Silhouette* das Leerwort enthält.

(c) Wir geben eine Ableitung des Wortes `//\/_\_\` in der Grammatik an.

$$\begin{aligned}
\mathit{Silhouette} &\Rightarrow \mathit{Berg}\\
&\Rightarrow \texttt{"/"}\ \mathit{Silhouette}\ \texttt{"\textbackslash"}\\
&\Rightarrow \texttt{"/"}\ \mathit{Berg}\ \mathit{Berg}\ \mathit{Plateau}\ \texttt{"\textbackslash"}\\
&\Rightarrow_p \texttt{"//"}\ \mathit{Silhouette}\ \texttt{"\textbackslash"/""}\ \mathit{Silhouette}\ \texttt{"\textbackslash_\textbackslash"}\\
&\Rightarrow_p \texttt{"//\textbackslash/"}\ \mathit{Plateau}\ \texttt{"\textbackslash_\textbackslash"}\\
&\Rightarrow_p \texttt{"//\textbackslash/_\textbackslash_\textbackslash"}
\end{aligned}$$

(d) Nein, $\mathcal{S}$ ist keine reguläre Sprache, da jeder Anstieg `/` irgendwann von einem Abstieg `\` gefolgt werden muss und die Zahl der Anstiege nicht beschränkt ist. Das Mitzählen der Anstiege erfordert einen unbeschränkten Speicher, die Sprache lässt sich daher nicht durch einen endlichen Automaten beschreiben, ist somit nicht regulär.