FGI-1 – Formale Grundlagen der Informatik I

Logik, Automaten und Formale Sprachen

Musterlösung 1 : Formale Sprachen und Automaten

Präsenzteil am 7.-10. April – Abgabe am 14.-17. April 2015

Präsenzaufgabe 1.1:

1. Wir betrachten das Alphabet $\Sigma = \{a, b, c\}$ und die Teilmengen $X, Y, Z \subseteq \Sigma^*$ mit $X = \{a, ab, \lambda\}$, $Y = \{c, bc, ac\}$ und $Z = \{c, cc\}$.

   (a) Bestimmen Sie $X \times Y$ und $|X \times Y|$.

   Lösung: $X \times Y = \{(a, c), (a, bc), (a, ac), (ab, c), (ab, bc), (ab, ac), (\lambda, c), (\lambda, bc), (\lambda, ac)\}$
   $|X \times Y| = |X| \cdot |Y| = 3 \cdot 3 = 9$.

   (b) Bestimmen Sie $X \cdot Y$ und $|X \cdot Y|$.

   Lösung: $X \cdot Y = \{ac, abc, aac, \underline{abc}, abbc, abac, c, bc, \underline{ac}\} = \{ac, abc, aac, abbc, abac, c, bc\}$
   Doppelte Einträge sind unterstrichen.
   $|X \cdot Y| = 7$

   (c) Bestimmen Sie $Z^0 \cup Z^1 \cup Z^2$.

   Lösung: Es ist

   $$\begin{aligned} Z^0 &= \{\lambda\} \\ Z^1 &= Z = \{c, cc\} \\ Z^2 &= Z \cdot Z = \{cc, ccc, cccc\} \end{aligned}$$

   also insgesamt

   $$Z^0 \cup Z^1 \cup Z^2 = \{\lambda, c, cc, ccc, cccc\}$$

   Hinweis: Die Notation Z^2 ist nicht ganz eindeutig, da wir sie sowohl für das kartesische Produkt $Z \times Z$ als auch für das Komplexprodukt $Z \cdot Z$ verwenden. Im Kontext von formalen Sprachen ist mit einem Exponenten aber typischerweise das Komplexprodukt $Z \cdot Z$ gemeint.

2. Sei Σ ein Alphabet und $A, B, C \subseteq \Sigma^*$ beliebige Sprachen. Beweisen oder widerlegen Sie folgende Gleichungen, indem Sie *zwei* (!) Inklusionsbeziehungen beweisen oder ein Gegenbeispiel angeben.

   (a) $(A \cup B)^* = A^* \cup B^*$

   Lösung: Gilt nicht. Wähle $A = \{x\}$ und $B = \{y\}$. Dann ist:

   $$\begin{aligned} (A \cup B)^* = \{x, y\}^* &= \{\lambda, x, y, xx, xy, yx, yy, xxx, \ldots\} \\ &\supsetneq A^* \cup B^* = \{x\}^* \cup \{y\}^* = \{\lambda, x, xx, \ldots\} \cup \{\lambda, y, yy, \ldots\} \end{aligned}$$

   Insbesondere ist z.B. $xy \in (A \cup B)^*$, aber $xy \notin A^* \cup B^*$.

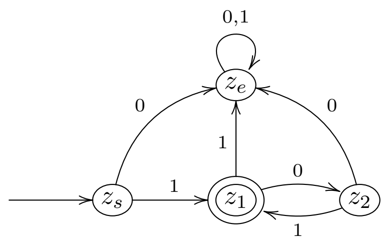
   (b) $(A \cup B) \cdot C = (A \cdot C) \cup (B \cdot C)$

Lösung: Wir zeigen zwei Inklusionen:

- Es gilt $(A \cup B) \cdot C \subseteq (A \cdot C) \cup (B \cdot C)$, denn wenn $w \in (A \cup B) \cdot C$, dann lässt sich w in xy zerlegen mit $x \in (A \cup B)$ und $y \in C$.
  Angenommen $x \in A$, dann ist $w = xy \in (A \cdot C)$; gilt dagegen $x \in B$, dann ist $w = xy \in (B \cdot C)$. Insgesamt also $w \in (A \cdot C) \cup (B \cdot C)$, was die Inklusion zeigt.
- Es gilt $(A \cdot C) \cup (B \cdot C) \subseteq (A \cup B) \cdot C$, denn wenn $w \in (A \cdot C) \cup (B \cdot C)$, dann ist $w \in (A \cdot C)$ oder $w \in (B \cdot C)$.
  Im ersten Fall lässt sich w in xy mit $x \in A$ und $y \in C$ zerlegen. Dann ist aber erst recht $x \in A \cup B$ und damit $w = xy \in (A \cup B) \cdot C$. Analog für $x \in B$. Dies zeigt die Inklusion.

Beide gezeigten Teilmengenbeziehungen zusammen beweisen die Mengengleichheit.

Präsenzaufgabe 1.2: Gegeben sei der folgende deterministische, endliche Automat A:



1. Geben Sie die formale Notation des DFA A an.

   Lösung: Es ist $A = (Z, \Sigma, \delta, z_s, Z_{end})$ mit der Zustandsmenge $Z = \{z_s, z_1, z_2, z_e\}$, dem Eingabealphabet $\Sigma = \{0, 1\}$, dem Startzustand z_s und der (einelementigen) Endzustandsmenge $Z_{end} = \{z_1\}$. Ferner ist die Übergangsfunktion δ gegeben durch $\delta(z_s, 0) = z_e$, $\delta(z_s, 1) = z_1$, $\delta(z_1, 0) = z_2$, $\delta(z_1, 1) = z_e$, $\delta(z_2, 0) = z_e$, $\delta(z_2, 1) = z_1$, $\delta(z_e, 0) = z_e$ und $\delta(z_e, 1) = z_e$.

2. Geben Sie die Rechnung auf dem Wort 10101 und die Rechnung auf dem Wort 10110 an. Welche dieser Rechnungen ist eine Erfolgsrechnung?

   Lösung: Es ist $(z_s, 10101) \vdash (z_1, 0101) \vdash (z_2, 101) \vdash (z_1, 01) \vdash (z_2, 1) \vdash (z_1, \lambda)$ und da z_1 ein Endzustand ist (und das Wort zu Ende gelesen wurde!) ist dies eine Erfolgsrechnung. (Diese Folge von Konfigurationen ergibt sich, indem man immer den Zustand einer Konfiguration und den ersten Buchstaben des noch zu lesenden Wortes nimmt und dies beides als Argument für *delta* benutzt, um den Nachfolgezustand zu ermitteln.)

   Ferner ist $(z_s, 10110) \vdash (z_1, 0110) \vdash (z_2, 110) \vdash (z_1, 10) \vdash (z_e, 0) \vdash (z_e, \lambda)$, was keine Erfolgsrechnung ist, da $z_e \notin Z_{end}$. (Hätte es keine 0-Kante an z_e gegeben, dann wäre die Rechnung in der Konfiguration $(z_e, 0)$ *blockiert*. Der Automat arbeitet dann nicht weiter und das Wort wird nicht akzeptiert - unabhängig davon ob z_e ein Endzustand ist oder nicht. Ein endlicher Automat akzeptiert ein Wort nur, wenn er das Wort *bis zu Ende liest* **und dann** *in einem Endzustand ist.*)

3. Emil Eilig behauptet, dass die vom Automaten akzeptierte Sprache $M = \{1\} \cdot (\{0\} \cdot \{1\})^*$ ist. *Beweisen* Sie, dass tatsächlich $L(A) = M$ gilt, indem sie *zwei Mengeninklusionen beweisen.*

   Lösung: Wir zeigen zwei Richtungen:

   (a) $M \subseteq L(A)$. Sei $w \in M$, dann hat w die Form $w = 1 \cdot (01)^n$ mit $n \in \mathbb{N}$. Wir müssen zeigen, dass A dieses Wort akzeptiert, dass also $w \in L(A)$ gilt. Nach Lesen der 1 ist A in z_1. Ist nun $n = 0$ (also $w = 1$), so sind wir fertig und A akzeptiert, da z_1 ein Endzustand ist. Ist $n > 0$, so überführt die 0 den Automaten A in den Zustand z_2 und die anschließende 1 ihn wieder in z_1. Ist das Wort nun gelesen (war also $n = 1$), so ist das Wort wieder zu Ende gelesen und wird akzeptiert, da z_1 ein Endzustand ist. Ansonsten wiederholt sich dies. Nach endlich vielen (nämlich n) Wiederholungen ist das Wort zu Ende gelesen, der Automat in z_1 und das Wort wird akzeptiert.

   (b) $L(A) \subseteq M$. Sei nun $w \in L(A)$, d.h. w ist ein Wort, das von dem Automaten akzeptiert wird (mehr wissen wir zunächst nicht). Wir müssen nun zeigen, dass daraus folgt, dass $w \in M$ gilt. Zunächst bemerken wir, dass w nicht mit 0 beginnen kann. Dies würde den Automaten in z_e überführen, einen Zustand, der nachfolgend nicht verlassen werden kann. w würde also nicht akzeptiert werden im Widerspruch zu $w \in L(A)$. w muss also mit 1 beginnen. Gilt nur $w = 1$ (was $w \in L(A)$ erfüllt), so gilt auch $w \in M$. Besteht w aus mehr als einem Buchstaben, so bemerken wir zunächst, dass nun keine 1 folgen kann, da der Automat dann wieder in z_e wäre. D.h. w muss mit 10 beginnen. Der nächste Buchstabe kann wieder keine 0 sein, da uns dies wieder in z_e überführt, muss also eine 1 sein. Das Wort 101 akzeptiert der Automat nun und dieses ist auch in M. Ist das Wort länger, so kann obige Argumentation wiederholt werden und insgesamt folgt damit aus $w \in L(A)$ auch $w \in M$.

von
4

Übungsaufgabe 1.3: Gegeben die formalen Sprachen $L_1 = \{0^i \mid i \in \mathbb{N}\} = \{0\}^*$, $L_2 = \{1^i \mid i \in \mathbb{N}\} = \{1\}^*$ und $L_3 = \{(01)^i \mid i \in \mathbb{N}\} = \{01\}^*$ über dem Alphabet $\Sigma = \{0, 1\}$. Berechnen Sie die nachfolgenden Mengen. Geben Sie Ihr Ergebnis dabei möglichst kurz an. So ist z.B. $\{0\}$ besser als $\{0, 1\} \setminus \{1\}$. (Geben Sie hier lediglich das Ergebnis an, ein strenger Gleichheitsbeweis ist nicht nötig. Sie können Ihr Ergebnis aber begründen, wenn es Ihnen nötig erscheint.)

1. $L_3 \cap \Sigma^*$

   Lösung: $L_3 \cap \Sigma^* = L_3$, da $L_3 \subseteq \Sigma^*$ gilt.

2. $L_3 \cup \Sigma^*$

   Lösung: $L_3 \cup \Sigma^* = \Sigma^*$, da wieder $L_3 \subseteq \Sigma^*$ gilt.

3. $L_1 \cap L_2$

   Lösung: $L_1 \cap L_2 = \{\lambda\}$, denn das einzige Element, das L_1 und L_2 gemeinsam haben, ist λ.

4. $L_3 \setminus L_1$

   Lösung: $L_3 \setminus L_1 = \{01\}^+$, da lediglich das Wort $0^0 = \lambda$ aus L_1 auch in L_3 enthalten ist (da $(01)^0 = \lambda$). Wenn man dieses nun aus L_3 entfernt, bleibt $\{01\}^+$ übrig.

5. $\Sigma^* \setminus L_1$

   Lösung: $\Sigma^* \setminus L_1 = \{w \in \Sigma^* \mid w \text{ enthält (mindestens) eine } 1\}$, denn in L_1 sind genau jene Worte, die nur aus 0en bestehen. Entfernt man diese aus Σ^*, so bleiben nur Worte übrig, die mindestens eine 1 enthalten.

6. $(L_3 \cup L_1) \cap \Sigma^4$

   Lösung: $(L_3 \cup L_1) \cap \Sigma^4 = \{0101, 0000\}$, da in Σ^4 genau jene Worte aus 0en und 1en enthalten sind, die aus vier Symbolen bestehen. In L_3 ist dies gerade das Wort 0101 und in L_1 das Wort 0000.

7. $\{0\} \cdot \Sigma^* \cdot \{1\}$

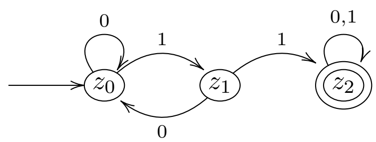
   Lösung: $\{0\} \cdot \Sigma^* \cdot \{1\} = \{w \in \Sigma^* \mid w \text{ beginnt mit 0 und endet mit } 1\}$. Man beachte, dass $\{0\} \cdot \Sigma^* \cdot \{1\} \subsetneq \Sigma^*$ gilt, da in Σ^* bspw. auch Wörter sind, die mit 1 beginnen, die Worte in $\{0\} \cdot \Sigma^* \cdot \{1\}$, aber zwingend mit 0 beginnen müssen.

8. $L_1 \cdot \Sigma^* \cdot L_2$

   Lösung: $L_1 \cdot \Sigma^* \cdot L_2 = \Sigma^*$, da anders als eben nun jedes Wort aus Σ^* auch in $L_1 \cdot \Sigma^* \cdot L_2$ liegt. Sei nämlich $w \in \Sigma^*$, so ist $w = \lambda \cdot w \cdot \lambda \in L_1 \cdot \Sigma^* \cdot L_2$. Die Beschreibung "in $L_1 \cdot \Sigma^* \cdot L_2$ sind jene Worte, die mit beliebig vielen 0en beginnen und mit beliebig vielen 1en enden" ist zwar auch richtig, kann aber wie oben viel kompakter formuliert werden (und ist zudem irreführend).

von
4

Übungsaufgabe 1.4: Gegeben sei der folgende deterministische, endliche Automat A, für den wir die akzeptierte Sprache bestimmen wollen.



Emil Eilig meint wieder, die Lösung zu kennen. Er behauptet $L(A) = \{0,1\}^* \cdot \{11\} \cdot \{0,1\}^* =: M$. Zeigen Sie, dass dies tatsächlich stimmt.

Hinweis: Achten Sie besonders darauf, die zwei Richtungen beim Beweis der Mengengleichheit sauber zu trennen. Werfen Sie ggf. noch einmal einen Blick in die Lösung der Präsenzaufgaben.

Lösung: Wir zeigen die zwei Teilmengenbeziehungen $L(A) \subseteq M$ und $M \subseteq L(A)$.

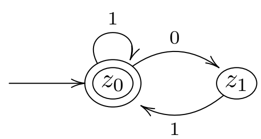
1. $L(A) \subseteq M$. Sei $w \in L(A)$. Damit A ein Wort akzeptiert, muss die Rechnung in z_2 enden. z_2 kann aber nur durch das Lesen zweier direkt aufeinanderfolgender 1en erreicht werden. Da der Automat insgesamt nur 0 oder 1 lesen kann folgt daraus, dass sich das Wort w zerlegen lässt in $w = x_1 \cdot x_2 \cdot x_3$ mit $x_2 = 11$ und $x_1, x_3 \in \{0,1\}^*$, woraus $w \in M$ folgt. (Achtung: Wir sagen, dass sich w so zerlegen lässt, nicht dass jedes Wort, dass diese Darstellung hat auch von A akzeptiert wird. Das folgt erst im nächsten Teilschritt. Theoretisch könnte es an dieser Stelle sein, dass nicht jedes $x_1 \in \{0,1\}^*$ bei der Zerlegung von w auftritt. Dies würde obige Argumentation aber nicht zerstören! Z.B. könnte man die 0-Kante von z_1 nach z_0 und die 1-Kante von z_2 zu sich selbst streichen und trotzdem wäre die Argumentation hier richtig! Dass x_1 dann genauer betrachtet sogar in $\{0\}^*$ liegt und nicht bloß in $\{0,1\}^*$, ist kein Widerspruch, da ja $\{0\}^* \subseteq \{0,1\}^*$ ist. Insgesamt wäre ja auch tatsächlich noch $L(A) \subseteq M$. Nur die andere Richtung würde dann nicht mehr klappen!)

2. $M \subseteq L(A)$. Sei $w \in M$, dann lässt sich w zerlegen in $w = x_1 \cdot x_2 \cdot x_3$ mit $x_1, x_3 \in \{0,1\}^*$ und $x_2 = 11$. Da in w das Teilwort 11 auftritt, muss dieses Teilwort auch an einer Stelle *das erste Mal* auftreten. Somit lässt sich w sogar zerlegen in $w = x_1' \cdot x_2' \cdot x_3'$ mit $x_2' = 11$, $x_3' \in \{0,1\}^*$ und $x_1' \in \{0,1\}^*$, wobei in x_1' das Teilwort 11 *nicht* auftritt und x_1' auch nicht auf 1 endet (dann könnte x_2' nämlich einen Buchstaben vorher beginnen!). Wird nun das Teilwort x_1' gelesen, so überführt dies den Automaten in den Zustand z_0 (vgl. die nachfolgenden Übungsaufgabe, wo im Prinzip der Teilautomat aus z_0 und z_1 konstruiert wird, nur dass die 0 und die 1 vertauscht werden). Das Teilwort $x_2' = 11$ überführt den Automaten in z_2 und das Teilwort x_3' belässt ihn dann dort. w kann also bis zum Ende gelesen werden und A ist dann im Zustand z_2. Da z_2 ein Endzustand ist, wird das Wort akzeptiert, also gilt $w \in L(A)$.

von
4

Übungsaufgabe 1.5: Sei $M = ((\{0\} \cdot \{1\}) \cup \{1\})^*$. Konstruieren Sie einen deterministischen, endlichen Automaten A mit $L(A) = M$ und beweisen Sie, dass dies tatsächlich gilt.

Lösung: Sei der deterministische, endliche Automat A durch folgendes Schaubild gegeben:



Wir behaupten $L(A) = M$ gilt. Beweis:

1. $M \subseteq L(A)$. Sei $w \in M$, dann lässt sich w zerlegen in $w = x_1 x_2 x_3 \dots x_n$, mit $n \in \mathbb{N}$ und $n > 0$ (im Spezialfall $w = \lambda$ gilt $w \in M$ und auch $w \in L(A)$, da der Start- auch ein Endzustand ist), wobei $x_i = 1$ oder $x_i = 01$ für jedes i gilt. Der Automat ist nach Lesen jedes x_i wieder im Zustand z_0. Damit ist er auch nach Lesen von x_n und damit nach Lesen von w in z_0 und es gilt, da z_0 ein Endzustand ist, $w \in L(A)$.

2. $L(A) \subseteq M$. Sei $w \in L(A)$. Zunächst kann $w = \lambda$ sein. In diesem Fall gilt aber auch $w \in M$. Ist nun $w \neq \lambda$, so beginnt w mit 0 oder 1. Ist im zweiten Fall das Wort bereits zu Ende, so ist also $w = 1$ (was $w \in L(A)$ nicht widerspricht). In diesem Fall ist aber auch $w \in M$. Beginnt w mit 0, so kann w nicht zu Ende sein, da $w \in L(A)$ gilt. Nach Lesen von 0 ist A in z_1, es kann also nur eine 1 gelesen werden (insgesamt also 01), was den Automaten wieder in z_0 überführt. Endet das Wort hier, ist also $w = 01$, so gilt auch $w \in M$. Da wir nun wie zu Anfang in z_0 sind, kann für längere Worte obige Argumentation auf das nun kürzere (Rest-)Wort angewandt werden. Insgesamt folgt damit, dass jedes $w \in L(A)$ auch in M ist.

A akzeptiert also genau die Worte aus M, was gerade jene Worte aus 0en und 1en sind, die keine zwei aufeinanderfolgenden 0en enthalten.

Informationen und Unterlagen zur Veranstaltung unter:

`http://www.informatik.uni-hamburg.de/TGI/lehre/vl/SS15/FGI1`