TI Übungsstunde 5

Marcel Schmid marcesch@student.ethz.ch

23.09.2020

---

1 Korrekturen

- Lemma 3.3 Beweise sehr, sehr gut! Genau so machen an einer Prüfung
- Aufpassen, mit "gefährlichen" Schlüssen, e.g. wenn L_1 nicht regulär ist, dann ist $L_1 \cup L$ nicht regulär für L regulär

⇒ "Sinn" der Aufgabe 12, solche intuitiven Annahmen zu widerlegen!

- Nur in Vorlesung/Buch/Serien bewiesene Aussagen ohne Beweis weiterverwenden
- Bei Aufgabe 12 waren viele fast zu lasch mit Begründung/Beweise der Nichtregularität ("offensichtlich")

⇒ E.g. $\{1^n 0^n \mid n \in \mathbb{N}\} \notin \mathbb{L}_{EA}$ ok nur mit Begründung

⇒ Aber bspw. $\{0^i 1^j \mid i \geq j\}$ muss dann gezeigt werden

- Ein EA ist ein Quintupel, keine Menge!

2 Theorie/Repetition

2.1 Nichtdeterministische EA

- Grosser Unterschied zwischen EA und NEA?

⇒ $\delta_D : Q \times \sigma \to Q$ wird zu $\delta_N : Q \times \Sigma \to \mathcal{P}(Q)$

⇒ Also kann es nun mehrere oder keine Transitionen für ein $a \in \Sigma$ geben!

⇒ Der NEA ist eine Art "intelligente" Maschine, der automatisch/magisch den richtigen Berechnungspfad auswählt.

- Die Konstruktion aus Satz 3.2. zeigt, dass NEAs und DEAs die gleichen Sprachen erkennen (nichtdeterminismus ist kein "stärkeres" mathematisches Modell)

2.2 Turing Machines

- Ihr werdet nie eine TM formell angeben müssen, aber ihr solltet das Modell kennen
- Unterschied zu EAs?

⇒ "Arbeitsspeicher" mit separatem Alphabet!

- $q_{\text{accept}}, q_{\text{reject}}$ und unendliche Berechnung

⇒ $L(M)$ sind alle Eingaben von M, wo M in einem akzeptierenden Zustand landet

⇒ Damit sind alle akzeptierten Berechnungen endlich

⇒ man muss daher aufpassen mit Aussagen über $L(M)^C$

- $\mathcal{L}_{RE} = \{L(M) \mid M \text{ ist eine TM}\}$, rekursiv aufzählbar
- $\mathcal{L}_R = \{L(M) \mid M \text{ hält zudem immer}\}$, rekursiv

- MTMs: wie in Section 4.4. gesehen kann man immer eine MTM konstruieren statt einer TM (L4.2) $\Rightarrow$ das macht Beweise der Form $L \in L_{R/RE}$ leichter
- NTM wieder sehr ähnlich wie NEAs; die Maschine macht "automatisch" das richtige, i.e. trifft die richtige Wahl
- Kod(M): es gibt die Möglichkeit, eine TM eindeutig mit endlich vielen Bits zu kodieren.

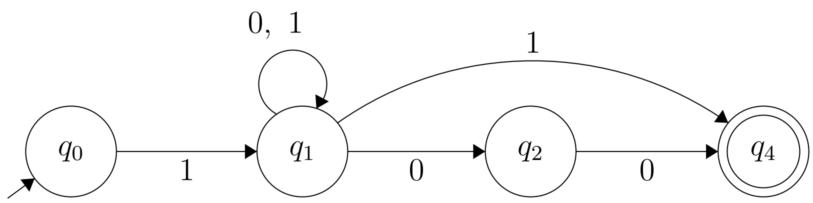
3 Übungen

3.1 HS8, 1a)

Sei $L = \{1x \mid x = y1 \text{ für ein } y \in \{0,1\}^* \text{ oder } x = z00 \text{ für } z \in \{0,1\}^*\}$. Konstruiere einen NEA mit höchstens 4 Zuständen, der L erkennt.

1. Wir brauchen sicher schon mal 2 States, um die Bedingung $1x$ zu Prüfen.
2. Weiter brauchen wir sicherlich 2 weitere States, um die Bedingung $x = z00$ zu prüfen.
3. Der Trick besteht nun darin, zu sehen, wie wir noch die letzte Bedingung $x = y1$ prüfen können: wir haben keine States mehr "über", die wir hinzufügen könnten. Aber wir können die 4 Zustände noch um Transitions ergänzen.

   $\Rightarrow$ Von den 4 Zst. ist lediglich der Zustand "hinter" $x00$ akzeptierend. Wir wollen, dass ein Suffix 1 auch dort endet $\Rightarrow$ daher können wir eine Transition dorthin noch einfügen, welche bei "1" genommen werden kann:

4. Begründung: Wir müssen begründen, dass alle Wörter, welche im akzeptierenden Zustand enden, von der gefrgten Form sind (Präfix und Suffix wird geprüft).

   Zudem sollten wir kurz erklären, warum ein Wort in L erkennt werden kann ($y/z \in \{0,1\}^*$ wird in q_1 geprüft.)

3.2 HS18, 3b))

Sei $L_n = \{x \in \{0,1\}^* \mid |x|_1 \geq n\}$. Zeige, dass jeder DEA, der L_n akzeptiert, mindestens $n+1$ Zst. hat:

1. Das ist sehr ähnlich wie die üblichen Lemma 3.3. Beweise, aber wenn mans noch nie gesehen hat, kanns schwierig sein.
2. Wir machen wie immer einen Widerspruchsbeweis: Wir nehmen an, dass es einen EA $A = (Q, \Sigma, \delta, q_0, F)$ gibt, der weniger als n Zustände braucht und L_n akzeptiert.
3. Dann betrachten wir die folgenden Wörter:
   $$1^i, \quad i \in \{0, \dots, n\}$$
4. Für unser EA muss Lemma 3.3 gelten, in anderen Worten: Falls für $i \neq j$ gilt $\hat{\delta}(q_0, 1^i) = \hat{\delta}(q_0, 1^j)$, dann gilt für alle $z \in \Sigma^*$:
   $$\hat{\delta}(q_0, 1^i z) = \hat{\delta}(q_0, 1^j z)$$
5. Da wir per Annahme nur n Zustände haben, aber $n+1$ Wörter betrachten, muss es i, j geben mit $i < j$ so dass obige Gleichung erfüllt ist.

$\Rightarrow$ das führt aber direkt zu einem Widerspruch: denn für das Suffix $z = 1^{n-i}$ gilt:

$$1^i z = 1^i 1^{n-i} = 1^n \in L_n$$

Aber:

$$1^j z = 1^j 1^{n-i} = 1^{n-i+j} \notin L_n$$

Denn wegen $i < j$ folgt, dass die Anzahl an Einsen in $1^j z$ strikt grösser als n ist.

6. Somit haben wir einen Widerspruch und wir haben die Behauptung gezeigt.

3.3 HS15, 4a)

Zeige: $L = \{0^{n\cdot\lceil\sqrt{n}\rceil} \mid n \in \mathbb{N}\}$ ist nicht regulär (mit Kolmogorov-Methode):

1. Wir betrachten die folgende Präfixsprache für alle $n \in \mathbb{N}$:

$$L_{0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil}}$$

2. Note: Vor einer Woche war λ das erste Wort in der Sprache und basierend auf dieser Begründung konnten wir das nächste Wort in der Präfixsprache in kanonischer Ordnung suchen.

   $\Rightarrow$ Hier "sehen" wir, dass $0^{\lceil\log n\rceil}$ das erste Wort der Sprache ist, aber ist es das wirklich?

   $\Rightarrow$ Wir müssen zeigen, dass für alle $n \in \mathbb{N}$ (oder zumindest für unendlich viele, siehe später im Beweis) das erste Wort in $L_{0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil}}$ $0^{\lceil\log n\rceil}$ ist.

3. Dazu schauen wir uns das folgende Wort an:

$$w := 0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil} 0^{\lceil\log n\rceil} = 0^{n\cdot\lceil\sqrt{n}\rceil} \in L$$

   Offenbar ist das in L, also gibt es ein $k \in \mathbb{N}$, so dass w das k-te Wort in kanonischer Ordnung in L ist. Wir schreiben $w = w_k$

4. Jetzt schauen wir uns das $(k-1)$-te Wort in L an:

$$w_{k-1} = 0^{(n-1)\cdot\lceil\sqrt{n-1}\rceil}$$

   da $n \cdot \lceil\sqrt{n}\rceil$ offenbar monoton steigt.

5. Unser Ziel ist es jetzt zu zeigen, dass quasi kein Wort "Platz" hat zwischen w_{k-1} und w_k: Nehmen wir mal an, dass w_k nicht das erste Wort in $L_{0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil}}$ ist. Dann gäbe es ein Wort $w_x = 0^x$ so dass $0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil} 0^x = 0^{m\cdot\lceil\sqrt{m}\rceil}$ gilt für ein $m < n$.

6. Doch für $x = 0$ (i.e. $0^x = \lambda$) kriegen wir das Folgende[1]:

   Claim: $(n-1)\cdot\lceil\sqrt{n-1}\,\rceil < n \cdot \lceil\sqrt{n}\rceil - \lceil\log n\rceil + x$

   Proof:

$$\begin{aligned}
&(n-1)\cdot\lceil\sqrt{n-1}\,\rceil < n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil + x \\
\iff\ & n\cdot\lceil\sqrt{n-1}\,\rceil - \lceil\sqrt{n-1}\,\rceil \le n\cdot\lceil\sqrt{n}\,\rceil - \lceil\sqrt{n}\,\rceil < n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil \\
\iff\ & -\lceil\sqrt{n}\,\rceil < -\lceil\log n\rceil
\end{aligned}$$

   Was offensichtlich für alle $n \ge n_0$ gilt für ein $n_0 \in \mathbb{N}$.

7. Was haben wir jetzt gezeigt?

   $\Rightarrow$ Wenn es ein Wort w_x in $L_{0^{n\cdot\lceil\sqrt{n}\rceil - \lceil\log n\rceil}}$ geben sollte, welches eine kleinere kanonische Ordnung hat als $w = w_k$, dann müsste es zwischen w_{k-1} und w_k liegen in L

   $\Rightarrow$ Widerspruch, also ist $0^{\lceil\log n\rceil}$ für alle $n \ge n_0$ (sprich unendlich viele) das erste Wort in der Präfixsprache

8. Somit kriegen wir aber einen Widerspruch zu Satz 3.1: Gemäss dem müsste dann nämlich für alle $n \ge n_0$ gelten, dass

$$K(0^{\lceil\log n\rceil}) \le \log(1+1) + c, \quad c \text{ konst.}$$

   Was natürlich nicht sein kann. Somit ist unsere Annahme, dass L regulär ist, falsch.

[1] Anm.: wir müssen diesen Claim für alle $0 \le x \le \lceil\log n\rceil$ zeigen. Wir können uns aber auf $x = 0$ beschränken, da wir zeigen wollen, dass wir mit keinem Suffix zu einem kleineren Wort als w_{k-1} "gelange" können – wenn wir nicht einmal mit dem leeren Wort ein kleineres Wort erreichen, dann erübrigen sich die anderen Suffixe damit auch.

4 Neue Serie

- 20 Punkte maximal möglich – die Bonusaufgabe zählt wieder nicht
- Bonusaufgabe a) ist "normal" schwierig, b) ist recht tricky
- Bei Pumping-Lemma aufpassen, was ihr wählen dürft und was ihr für alle Möglichkeiten zeigen müsst.