Einführung in die Theoretische Informatik

Tutorium – Woche 4

Esther Ney

Sommersemester 2025

Organisatorisches

Zulip

Für die diesjährige THEO-Vorlesung wurden **inoffizielle** Streams auf der ZULIP-Instanz der TUM INFORMATIK erstellt. Dieser wird von den meisten Tutor*innen, aber nicht von PROF. ALBERS gelesen; wir versuchen natürlich trotzdem, organisatorische Fragen bei Bedarf weiterzuleiten. Man kann mich dort auch in Direktnachrichten anschreiben.

- Tutorin: ESTHER NEY
- Tutorien: Dienstag 16-18 (Di-16-2) und Donnerstag 16-18 (Do-16-1)
- E-Mail: `esther.ney@tum.de`
- Aufbau einer Tutorstunde: kurze Wiederholung wichtiger Definitionen und Lemmata, dann selbständiges Bearbeiten der Aufgaben
- Folien: auf ZULIP oder `https://reflect.moe/uni/theo25/`

Disclaimer

Hinweis

Diese Folien sind keine offizielle Musterlösung und dienen lediglich als zusätzliches Material für die Nachbereitung der Tutorien.

Tutorienlösungen können fehlerhaft sein.

Im Zweifel gilt immer das, was in den offiziellen Vorlesungsfolien und in der Musterlösung steht.

Urheberrechtlicher Hinweis

Satz 3.23 – NFA zu RE (Folie 63)

Ein zu einem NFA N äquivalenter RE r ist gegeben mit folgendem Algorithmus.

(a) Preprocessing: Füge ggf. einen neuen Startzustand und einen neuen Endzustand mit ϵ-Übergängen zum Start- bzw. von Endzuständen hinzu.

(b) Wende folgende Regeln solange wie möglich an.

- Ersetze (q, α_1, p) und (q, α_2, p) mit $(q, \alpha_1 \mid \alpha_2, p)$. Danach:
- Wähle Zustand q beliebig. Gilt (q, σ, q), ersetze alle (p, α, q) und (q, β, r) (wobei gilt $p \neq q$ oder $r \neq q$) mit $(p, \alpha\sigma^*\beta, r)$. Falls es kein σ mit (q, σ, q) gibt, ersetze stattdessen (p, α, q) und (q, β, r) mit $(q, \alpha\beta, r)$.
- Lösche q mit allen ein- und ausgehenden Übergängen und fahre mit einem neuen Zustand fort, bis nur noch Start- und Endzustand bleiben.

(c) Lese den RE r vom Übergang von Start- zu Endzustand ab.

Ardens Lemma

Satz 3.28 – Ardens Lemma

Sind α, β und X reguläre Ausdrücke mit $\varepsilon \notin L(\alpha)$, so gilt

$$X \equiv \alpha X \mid \beta \implies X \equiv \alpha^* \beta$$

Das brauchen wir gleich!

GAUSS-Algorithmus

GAUSS-Algorithmus – ab Folie 70

Ein zu einem NFA N äquivalenter RE r ist gegeben mit folgendem Algorithmus.

(a) Preprocessing: Füge ggf. einen neuen Startzustand und einen neuen Endzustand mit ϵ-Übergängen zum Start- bzw. von Endzuständen hinzu.

(b) Stelle ein Gleichungssystem auf: Für alle $q \in Q$ gilt $q \equiv \alpha p$ genau dann, wenn $q \in \delta(p, \alpha)$.

(c) Wende folgende Regeln solange wie möglich an.
   - Wähle Zustand q beliebig. Falls $q \equiv \alpha q \mid \beta$, ersetze durch $q \equiv \alpha^* \beta$ mit ARDENS Lemma, solange wie möglich.
   - Ersetze alle Vorkomnisse von q in anderen Gleichungen mit der oben gewonnenen Gleichung.
   - Lösche die Gleichung für q und fahre mit einem neuen Zustand fort, bis nur noch Start- und Endzustand bleiben.

(d) Lese den RE r vom Übergang von Start- zu Endzustand ab.

Pumping-Lemma

Intuition: Wenn L eine reguläre Sprache ist, gibt es einen DFA, der L darstellt. Manchmal können wir zeigen, dass für alle möglichen DFAs zu L ein **Widerspruch** entsteht.

Satz 3.37 – Pumping-Lemma

Sei L eine reguläre Sprache. Dann gibt es ein $n \geq 1$, dass alle $z \in L$ mit $z \geq n$ eine Zerlegung $z \equiv uvw$ besitzen, für die gilt:

- $v \neq \epsilon$
- $|uv| \leq n$
- $\forall i \geq 0.\ uv^i w \in L$

Intuitiv ist n die Anzahl der Zustände eines möglichen DFAs.

Es gibt Sprachen, die nicht regulär sind, aber das Pumping-Lemma “erfüllen”. Satz 3.60 (in der Literatur bekannt als der Satz von MYHILL und NERODE) liefert uns eine genau-dann-wenn-Bedingung dafür, dass eine Sprache regulär ist. Ein Beispiel hierfür ist auf der Wikipedia (Link) zu finden.

Äquivalenzproblem

Können wir mit Axiomen herleiten, ob zwei reguläre Ausdrücke äquivalent sind?

Satz 3.34 – REDKOs Satz 1964

nö

Aber wir können indirekt überprüfen, ob zwei Automaten bzw. reguläre Ausdrücke äquivalent sind!

Satz 3.48 – DFA-Äquivalenz ist entscheidbar

Aber für DFAs können wir einen Automaten konstruieren, der genau dann leer ist, wenn $L(D_1) \neq L(D_2)$. Und das ganze geht in $O(|Q_1||Q_2||\Sigma|)$

Korollar 3.49 und Korollar 3.50 – NFA- und RE-Äquivalenz ist entscheidbar

Aus Satz 3.48 gewinnen wir einen exponentiellen Algorithmus, der prüft, ob zwei NFAs oder zwei REs äquivalent sind.

Vorbereitung

Überprüfen Sie, dass Sie mit den folgenden Begriffen vertraut sind.

- Ardens Lemma
- Pumping-Lemma
- Äquivalenzproblem

Aufgabe T 4.1

Gegeben sei der NFA N mit folgender graphischer Darstellung.

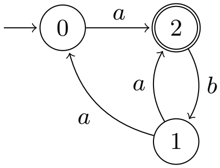

Abbildung 1 Der Automat N.

Bilden Sie einen regulären Ausdruck r mit $L(r) = L(N)$. Nutzen Sie hierfür den Algorithmus aus der Vorlesung (Satz 3.23) und eliminieren Sie die Zustände in *aufsteigender* Reihenfolge, d.h. eliminieren Sie zuerst 0, dann 1, dann 2.

Aufgabe T 4.1

Aufgabe T 4.2

Gegeben sei der DFA $M := (\{q_1, q_2, q_3\}, \{a, b\}, \delta, q_1, \{q_3\})$ mit folgender graphischer Darstellung.

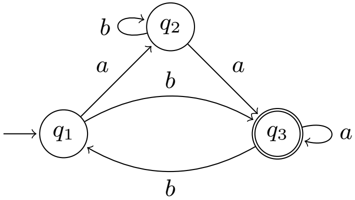

Abbildung 2 Der Automat M.

Berechnen Sie mit dem GAUSS-Verfahren (ab Folie 70) und ARDENs Lemma (Satz 3.28) einen regulären Ausdruck α mit $L(\alpha) = L(M)$.

Aufgabe T 4.2

Aufgabe T 4.3

Gegeben seien die DFAs M_1 und M_2 mit folgender graphischer Darstellung.

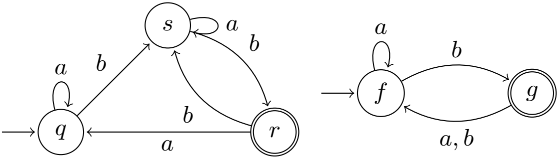

Abbildung 3 Die Automaten M_1 und M_2, wobei M_1 anders als auf dem Blatt aussieht.

(a) Konstruieren Sie einen DFA M mit $L(M) = L(M_1) \cap L(M_2)$, indem Sie die Produktkonstruktion (Satz 3.36) verwenden. Gibt es einen DFA M' mit $L(M) = L(M)$, der weniger Zustände als M hat?

(b) Konstruieren Sie nun einen Automaten für $L(M_1) \cup L(M_2)$.

Aufgabe T 4.3

Aufgabe T 4.4

Zeigen Sie für die Sprachen L_1, L_2 und L_3, dass sie NICHT regulär sind. Nutzen Sie hierfür das Pumping-Lemma.

(a) $L_1 := \{w \in \{0,1\}^* \mid w = w^R\}$

(b) $L_2 := \{w \in \{0,1\}^* \mid |w|_0 \geq |w|_1\}$

(c) $L_3 := \{a^{2^i} \mid i \geq 0\}$

Aufgabe T 4.4

Endterm 2023 – Aufgabe 1f), 3 Punkte

Wenn $A, B \subseteq \Sigma^*$ regulär sind und $\varepsilon \notin A$, dann sind alle Lösungen der Gleichung $X \equiv AX \cup B$ regulär.

Hinweis: Nutzen Sie die Abschlusseigenschaften regulärer Sprachen. Sie dürfen $X \equiv AX \cup B \implies X \equiv A^*B$ verwenden.

Altklausuraufgabe