universität innsbruck

Diskrete Mathematik

Ralph Bottesch
Burak Ekici
Johannes Koch
Georg Moser
David Obwaller
Vincent van Oostrom
Oleksandra Panasiuk

cbr.uibk.ac.at

Zusammenfassung der letzten LVA

Definition

Eine natürliche Zahl p heißt Primzahl, wenn $p \notin \{0, 1\}$ und p nur die trivialen Teiler besitzt.

Satz

Jede ganze Zahl größer als 1 kann als Produkt von Primzahlen geschrieben werden. Diese Primzahlen heißen Primfaktoren der Zahl und sind bis auf die Reihenfolge eindeutig bestimmt.

Satz (der kleine Satz von Fermat)

Sei p eine Primzahl und sei a eine ganze Zahl, die nicht von p geteilt wird. Dann gilt

$$a^{p-1} \equiv_p 1\,.$$

Inhalte der Lehrveranstaltung (cont'd)

Reguläre Sprachen

deterministische Automaten, nichtdeterministische Automaten, endliche Automaten mit Epsilon-Übergängen, reguläre Ausdrücke, Abgeschossenheit, Schleifenlemma

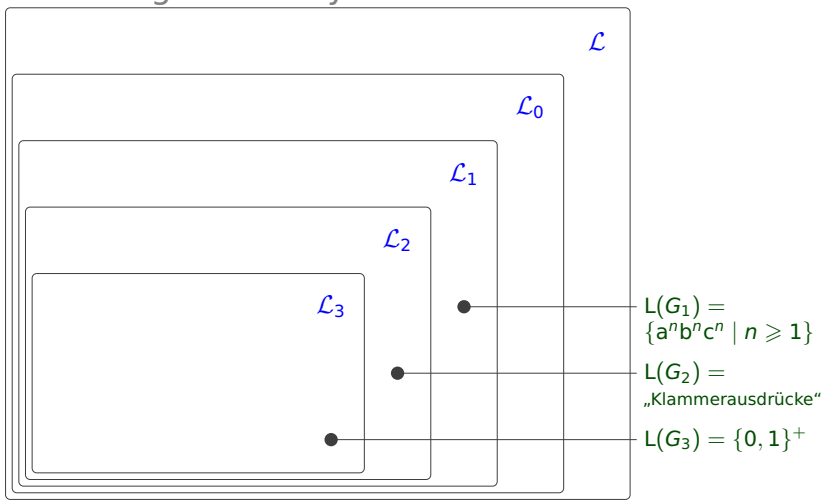
Berechenbarkeitstheorie

deterministische TM, nichtdeterministische TM, universelle TMs, Äquivalenzen

Komplexitätstheorie

Grundlagen, die Klassen P und NP, polynomielle Reduktionen, logspace Reduktionen

Zur Erinnerung: Chomsky-Hierarchie



Anwendungen von regulären Sprachen

- Software zum Entwurf und Testen von digitalen Schaltkreisen
- Softwarebausteine eines Compilers, etwa in der lexikalischen Analyse:
  1. lexikalische Scanner („Lexer“) wird mit endlichen Automaten implementiert
  2. Der lexikalische Scanner dient zur Aufteilung des Eingabetextes in logische Einheiten, wie Bezeichner oder Schlüsselwörter
- Software zum Durchsuchen umfangreicher Texte
- Software zur Verifizierung aller Arten von Systemen, die eine endliche Anzahl verschiedener Zustände besitzen
- Softwarebausteine eines Computerspiels:
  1. Kontrolle von Spielfiguren kann mit Hilfe eines endlichen Automaten implementiert werden
  2. erlaubt eine bessere Modularisierung des Codes

Beispiel

Alle endlichen Sprachen sind regulär.

Definition

Ein DEA ist gegeben durch ein 5-Tupel $A = (Q, \Sigma, \delta, s, F)$ sodass

1. Q eine endliche Menge von Zuständen
2. Σ eine endliche Menge von Eingabesymbole, (Σ wird auch Eingabealphabet genannt)
3. $\delta\colon Q \times \Sigma \to Q$ die Übergangsfunktion
4. $s \in Q$ der Startzustand
5. $F \subseteq Q$ eine endliche Menge von akzeptierenden Zuständen

Zu beachten: δ muss für alle möglichen Argumente definiert sein

Zustandstabelle

	$a_1 \in \Sigma$	$a_2 \in \Sigma$	$\cdots$
$q_1 \in Q$	$\delta(q_1, a_1)$	$\delta(q_1, a_2)$	$\cdots$
$q_2 \in Q$	$\delta(q_2, a_1)$		
$\vdots$	$\vdots$		

Zustandsgraph

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA, der (gerichtete) Zustandsgraph mit Startzustand s und akzeptierenden Zuständen F ist wie folgt definiert:

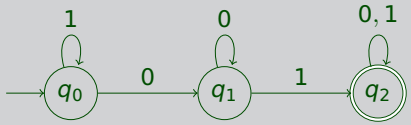
1. die Zustände sind die Ecken
2. die Kanten K sind
$$(p, q) \qquad p, q \in Q \text{ und } \exists a \in \Sigma \text{ mit } \delta(p, a) = q$$
3. die Kantenbeschriftung $b\colon K \to \Sigma$ ist definiert als
$$(p, q) \mapsto a \qquad \text{wenn } \delta(p, a) = q$$

Beispiel

Der DEA $A = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$ mit Zustandstabelle

	0	1
$\to q_0$	q_1	q_0
q_1	q_1	q_2
$*q_2$	q_2	q_2

besitzt den folgenden Zustandsgraphen:



Definition (erweiterte Übergangsfunktion)

Sei δ eine Übergangsfunktion, wir definieren die erweiterte Übergangsfunktion $\hat{\delta}\colon Q \times \Sigma^* \to Q$ induktiv:

$$\hat{\delta}(q, \epsilon) := q$$
$$\hat{\delta}(q, xa) := \delta(\hat{\delta}(q, x), a) \qquad x \in \Sigma^*,\ a \in \Sigma$$

Definition

Sei $A = (Q, \Sigma, \delta, q_0, F)$ ein DEA; die Sprache $\mathrm{L}(A)$ von A:

$$\mathrm{L}(A) := \{x \in \Sigma^* \mid \hat{\delta}(q_0, x) \in F\}$$

Satz (Erinnerung)

Sei A ein DEA, dann ist $\mathrm{L}(A)$ regulär und umgekehrt existiert zu jeder regulären Sprache L ein DEA A, sodass $L = \mathrm{L}(A)$

Beispiel

Für den obigen DEA A gilt $\hat{\delta}(q_0, 0010) = q_2$
Wir berechnen $\hat{\delta}(q_0, 0010)$ rekursiv wie folgt:

- $\hat{\delta}(q_0, 0010) = \delta(\hat{\delta}(q_0, 001), 0) = \delta(q_2, 0) = q_2$
- $\hat{\delta}(q_0, 001) = \delta(\hat{\delta}(q_0, 00), 1) = \delta(q_1, 1) = q_2$
- $\hat{\delta}(q_0, 00) = \delta(\hat{\delta}(q_0, 0), 0) = \delta(q_1, 0) = q_1$
- $\hat{\delta}(q_0, 0) = \delta(\hat{\delta}(q_0, \epsilon), 0) = \delta(q_0, 0) = q_1$

Beispiel

Für DEA A gilt $\mathrm{L}(A) = \{x01y \mid x, y \in \Sigma^*\}$. Die Sprache $\mathrm{L}(A)$ beschreibt die Menge aller Wörter, in welchen eine 1 direkt auf eine 0 folgt

Definition (alternativ)

Eine formale Sprache L heißt regulär, wenn $\exists$ DEA A, sodass $\mathrm{L}(A) = L$

Definition

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA; dann heißt der DEA $A' = (Q', \Sigma, \delta', s, F')$ der erreichbare Anteil von A, wobei

- $Q' = \{q \in Q \mid \exists x \in \Sigma^* \text{ mit } \hat{\delta}(s, x) = q\}$,
- $\delta'\colon Q' \times \Sigma \to Q'$ mit $\delta'(q, a) := \delta(q, a)$ für alle $q \in Q'$ und $a \in \Sigma$
- $F' = F \cap Q'$

Satz

Sei A ein DEA und A' der erreichbare Anteil von A, dann $\mathrm{L}(A) = \mathrm{L}(A')$

Beweisskizze.

- δ' ist wohldefiniert und $s \subseteq Q'$ und $F' \subseteq Q'$ per Definition
- also $(Q', \Sigma, \delta', s, F')$ ein DEA
- somit $\mathrm{L}(A) = \{x \in \Sigma^* \mid \hat{\delta}(s, x) \in F\} = \{x \in \Sigma^* \mid \hat{\delta}'(s, x) \in F'\} = \mathrm{L}(A')$

Beobachtung

Erreichbare Zustände können mittels Nachfolgersuche berechnet werden

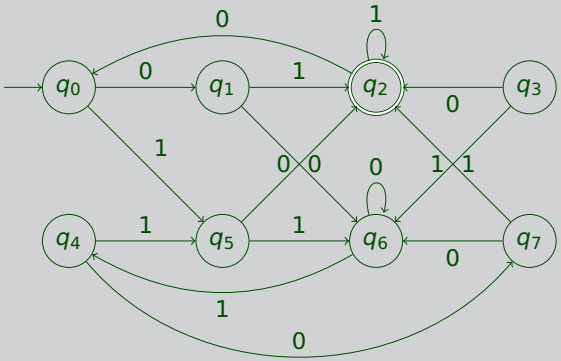
Definition (Nachfolgersuche für DEA)

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA

Setze $S := \{s\}$ und markiere s
Solange S nichtleer ist, wiederhole:
  Wähle eine Ecke e in S und entferne sie aus S
  Bestimme alle unmarkierten unmittelbaren Nachfolger von e
    markiere sie und gebe sie zu S dazu

Beispiel

Sei DEA B gegeben durch den Zustandsgraphen:



Erreichbarer Anteil erhält man duch Weglassen von q_3

Definition

Zwei Zustände $p \in Q$ und $q \in Q$ heißen äquivalent, wenn für jedes Wort $x \in \Sigma^*$, $\hat{\delta}(p,x) \in F$ genau dann, wenn $\hat{\delta}(q,x) \in F$

Beispiel (cont'd)

- Die Wörter ϵ, 0, und 1 sind nicht geeignet die Zustände q_0 und q_6 als nicht äquivalent festzustellen
- Aber $\hat{\delta}(q_0, 01) = q_2 \in F$ und $\hat{\delta}(q_6, 01) = q_4 \notin F$; also sind q_0 und q_6 nicht äquivalent

Satz

Äquivalenz von Zuständen ist eine Äquivalenzrelation

Beweis.

Mittels Lemma 2.11 ■

Definition

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA; zwei Zustände $p \in Q$ und $q \in Q$ sind unterscheidbar,

1. wenn $p \in F$ genau dann, wenn $q \notin F$ oder
2. wenn es unterscheidbare Zustände u und v sowie ein $a \in \Sigma$ gibt, sodass $\delta(p,a) = u$ und $\delta(q,a) = v$

Satz

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA und seien $p, q \in Q$. Dann sind p und q äquivalent, gdw sie nicht unterscheidbar sind

Beweis.

- $\Rightarrow$: wenn p und q unterscheidbar sind, sind sie nicht äquivalent
- $\Leftarrow$: sind Zustände p, q nicht äquivalent, dann sind sie unterscheidbar

wir zeigen $\Leftarrow$, die andere Richtung folgt leicht (siehe Skriptum)

Beweis von $\Leftarrow$.

betrachte $A = (Q, \Sigma, \delta, q_0, F)$

1. wir zeigen für alle $p, q \in Q$ und für alle $w \in \Sigma^*$
   Wenn $\hat{\delta}(p,w) \in F$ und $\hat{\delta}(q,w) \notin F$, dann sind p, q unterscheidbar.
2. strukturelle Induktion nach w
3. Basis $w = \epsilon$, dann sind p, q unterscheidbar nach Definition
4. Schritt $w = ax$, mit $r = \delta(p,a)$, $s = \delta(q,a)$
5. nach Voraussetzung gilt

$$\hat{\delta}(r,x) \in F \quad \hat{\delta}(s,x) \notin F$$

   mit IH sind r und s unterscheidbar
6. dann sind aber auch p, q unterscheidbar

■

Definition (Erinnerung)

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA. Zwei Zustände $p \in Q$ und $q \in Q$ sind unterscheidbar,

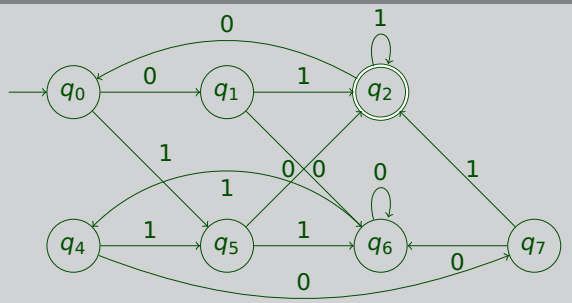
1. wenn $p \in F$ genau dann, wenn $q \notin F$ oder
2. wenn es unterscheidbare Zustände r und s sowie ein $a \in \Sigma$ gibt, sodass $\delta(p, a) = r$ und $\delta(q, a) = s$

Definition (Markierungsalgorithmus)

1. wenn p akzeptierend und q nicht, dann markiere $\{p, q\}$
2. sei a ein Eingabezeichen und
   - sei $\delta(p, a) = r$, $\delta(q, a) = s$
   - sodass $\{r, s\}$ markiert

   dann markiere $\{p, q\}$

Beispiel (cont'd)



q_0						
②	q_1					
①	①	q_2				
	②	①	q_4			
②	②	①	②	q_5		
③	②	①	③	②	q_6	
②		①	②	②	②	q_7

Markierungsalgorithmus liefert die folgenden Zustandsblöcke:

$\{q_0, q_4\}, \{q_1, q_7\}, \{q_5\}, \{q_2\}, \{q_6\}$

Definition (Minimierungsalgorithmus)

Sei $A = (Q, \Sigma, \delta, s, F)$ ein DEA

1. Bestimme den erreichbaren Anteil $A' = (Q', \Sigma, \delta', s, F')$ von A
2. Bestimme die äquivalenten Zustände von A'
3. Konstruiere den minimalen DEA $B = (Q_B, \Sigma, \delta_B, s_B, F_B)$
   - Q_B sind die Äquivalenzklassen von Q',
   - $\delta_B([q], a) := [p]$ wenn $\delta'(q, a) = p$ für $a \in \Sigma$ und $q \in Q'$,
   - der Startzustand ist $s_B := [s]$,
   - die akzeptierenden Zustände sind $F_B = \{[f] \mid f \in F'\}$.

Satz

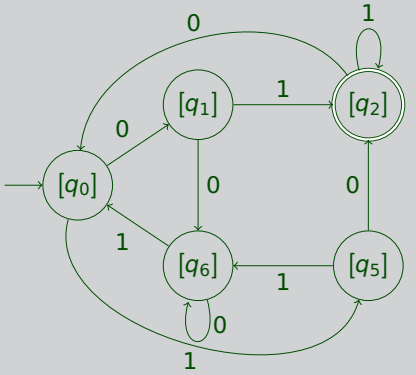
Sei A ein DEA und B der minimierte DEA:

1. $L(A) = L(B)$
2. *DEA B ist minimal*
3. *der minimale DEA eindeutig bis auf Umbenennung der Zustände*

■

Beispiel

Minimierung des DEA B: Zustand q_3 ist in A von q_0 aus nicht erreichbar; Zusammenfassen der äquivalenten Zustände des erreichbaren Anteils liefert einen minimalen Automaten



Definition

Ein nichtdeterministischer endlicher Automat (NEA) ist ein 5-Tupel $(Q, \Sigma, \delta, S, F)$, sodass

- Q eine endliche Menge von Zustände
- Σ eine endliche Menge, das Eingabealphabet; dessen Elemente Eingabezeichen genannt werden,
- die Zustandsübergangsfunktion

$$\delta : Q \times \Sigma \to \mathcal{P}(Q)$$

  gibt an wie sich der Zustand des Automaten bei einer Eingabe ändern kann
- $S \subseteq Q$, deren Elemente Startzustände genannt werden,
- $F \subseteq Q$, deren Elemente akzeptierende Zustände genannt werden

Beispiel

Gegeben der NEA N durch die Zustandstabelle

	0	1
$\to q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\{q_1\}$	$\{q_2\}$
$*q_2$	$\{q_2\}$	$\{q_2\}$

Definition

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NEA; dann heißt $\hat{\delta}\colon Q \times \Sigma^* \to \mathcal{P}(Q)$ die erweiterte Übergangsfunktion und ist wie folgt definiert:

1. $\hat{\delta}(q, \epsilon) := \{q\}$ für alle $q \in Q$
2. $\hat{\delta}(q, xa) := \bigcup_{p \in \hat{\delta}(q,x)} \delta(p, a)$ für alle $q \in Q$, $x \in \Sigma^*$ und $a \in \Sigma$

Beispiel

Wir berechnen $\hat{\delta}(q_0, 00101)$ für NEA N, schrittweise

- $\hat{\delta}(q_0, 00101) = \delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_2, 1) = \{q_0, q_2\}$
- $\hat{\delta}(q_0, 0010) = \delta(q_0, 0) \cup \delta(q_2, 0) = \{q_0, q_1, q_2\}$
- $\hat{\delta}(q_0, 001) = \delta(q_0, 1) \cup \delta(q_1, 1) = \{q_0, q_2\}$
- $\hat{\delta}(q_0, 00) = \delta(q_0, 0) \cup \delta(q_1, 0) = \{q_0, q_1\}$
- $\hat{\delta}(q_0, 0) = \delta(q_0, 0) = \{q_0, q_1\}$
- $\hat{\delta}(q_0, \epsilon) = \{q_0\}$

Definition

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NEA; dann wird

$$\mathrm{L}(N) := \{x \in \Sigma^* \mid \text{ es gibt einen Zustand } s \in S, \text{ sodass } \hat{\delta}(s, x) \cap F \neq \varnothing\}.$$

als die von N akzeptierte Sprache bezeichnet