TI Woche 6

~~Silvan Horvath~~ Robin Ebersberger

robin.ebersberger@inf.ethz.ch
Slides by Ramon Wick

31.10.2024

Overview

1. Theorie
   - Nichtdeterministische Turingmaschinen (NTM)
   - Countability
2. Üben

Nichtdeterministische Turingmaschinen

Nichtdeterminismus von Turingmaschinen

NTMs können "raten".
Normalerweise: NTM rät nichtdeterm. etwas und überprüft determ. ihre Wahl.
Beispiel: $L_{\neq} = \{x\#y \mid x, y \in \Sigma^*_{bool}\ , x \neq y\}$
Intuition: Wenn NTM etwas "richtiges" erraten kann, gibt es eine Berechnung, die das auch tut!

Berechnungsbaum einer NTM

Ein Berechnungsbaum $T_{M,x}$ einer NTM M auf $x \in \Sigma$ ist ein potentiell unendlicher, gerichteter Baum mit einer Wurzel.

- Jeder Knoten von $T_{M,x}$ ist mit einer Konfiguration beschriftet.
- Die Wurzel hat Eingangsgrad 0 und ist mit der Startkonfiguration beschriftet.
- Sei C die Konfiguration im Knoten p. Dann hat p genau so viele Kinder wie C Nachfolgekonfigurationen hat.

Fragen

- Warum ist er potenziell unendlich?

Vergleich: NEA Berechnungsbaum

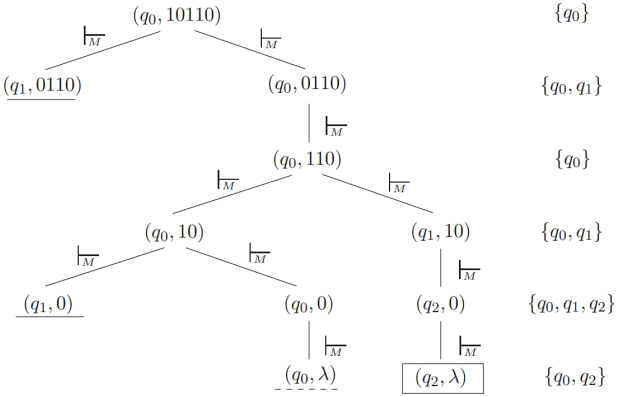


Figure: Abb. 3.16 aus dem Buch

NTM = TM

Satz 4.2

Sei M eine NTM. Dann existiert eine TM A, so dass:

1. $L(M) = L(A)$
2. falls M keine unendlichen Berechnungen auf Wörtern aus $L(M)^C$ hat, dann hält A immer.

Warum ist das so?

Kodierung von TMs

Jede TM kann als Bitstring dargestellt werden.
Die Kodierung von der TM M nennen wir $\text{Kod}(M)$

Die Menge aller Kodierungen KodTM ist definiert als:

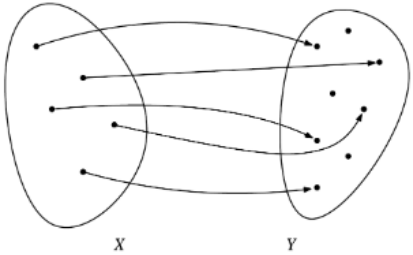
$$\text{KodTM} = \{\text{Kod}(M) \mid M \text{ ist eine TM}\}$$

Countability

Countability

Wir wollen zeigen, dass es Sprachen gibt, welche nicht von TMs akzeptiert werden können.
Wir wählen hier einen "Countability Approach".

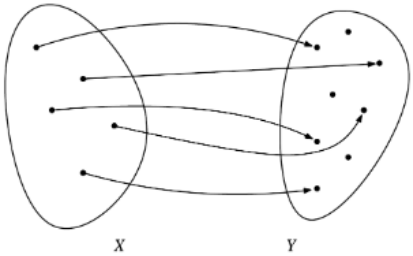
Recap



Def 5.1

Seien A und B zwei Mengen. Wir sagen, dass $|A| \leq |B|$, falls eine injektive Funktion f von A nach B existiert.

Recap

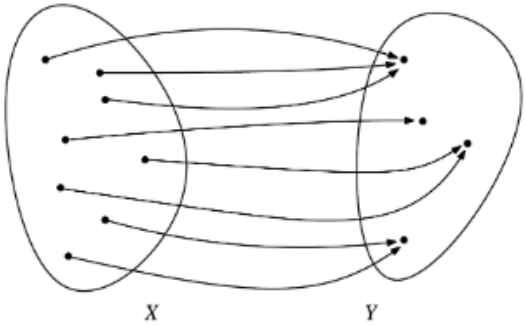


Def 5.1

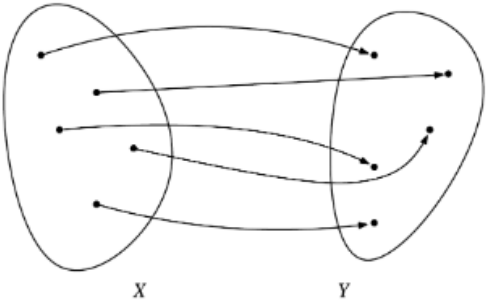
Seien A und B zwei Mengen. Wir sagen, dass $|A| \leq |B|$, falls eine injektive Funktion f von A nach B existiert.

Ist die Abbildung Kod : $M \mapsto \text{Kod}(M)$ injektiv? Was bedeutet "injektiv" nochmal?

Surjektion



Bijektion



Satz 5.1

Wir sagen, dass $|A| = |B|$, falls $|A| \leq |B|$ und $|B| \leq |A|$. Daher existiert eine Bijektion.

Abzählbarkeit

Def 5.2

Eine Menge A heisst **abzählbar**, falls A endlich ist oder $|A| = |\mathbb{N}|$

- Sei Σ ein beliebiges Alphabet. Ist es abzählbar?
- Ist Σ^* abzählbar?
- Ist die Menge $KodTM$ der Turingmaschinenkodierungen abzählbar?
- Seien A und B zwei abzählbare Mengen. Argumentiere, dass $A \cup B$ abzählbar ist.

Lemma 5.2

$(\mathbb{N} \setminus \{0\}) \times (\mathbb{N} \setminus \{0\})$ ist abzählbar.

Beweisidee

Unendliche 2-dimensionale Tabelle, so dass an der i-ten Zeile und j-ten Spalte, sich das Element $(i, j) \in (\mathbb{N} \setminus \{0\}) \times (\mathbb{N} \setminus \{0\})$ befindet.
Formal definiert man dabei die lineare Ordnung

$$(a, b) < (c, d) \iff a + b < c + d \text{ oder } (a + b = c + d \text{ und } b < d)$$

Abzählbarkeit

Lemma 5.2

$(\mathbb{N} \setminus \{0\}) \times (\mathbb{N} \setminus \{0\})$ ist abzählbar.

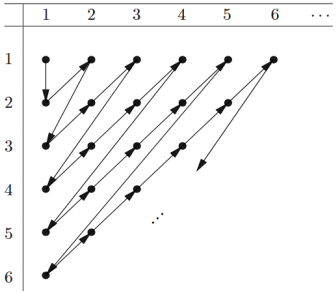


Figure: Abbildung 5.3 im Buch

Überabzählbarkeit

Satz 5.3

$[0, 1]$ ist nicht abzählbar.

Beweis mittels Diagonalisierung. Aufpassen auf 0 und 9, i.e. $1 = 0.\overline{9}$

Satz 5.4

Satz 5.4

$P(\Sigma^*_{bool})$ ist nicht abzählbar.

Korollar 5.1 - umformuliert

Da $KodTM \subseteq \Sigma^*_{bool}$ gilt: $|KodTM| < |P(\Sigma^*_{bool})|$ und somit existieren unendlich viele nicht rekursiv aufzählbare Sprachen über Σ_{bool}. Also Sprachen, die von keiner TM erkennt werden können.

Diagonalsprache

Sei M_i die i-te Turing Maschine. Wir definieren:

$$L_{diag} = \{w \in \Sigma^*_{bool} \mid w = w_i \text{ für ein } i \in N - \{0\} \text{ und } M_i \text{ akzeptiert } w_i \text{ nicht}\}$$

Satz 5.5

$L_{diag} \notin L_{RE}$. Daher es gibt keine TM, welche L_{diag} akzeptiert.

Sprachenklassen

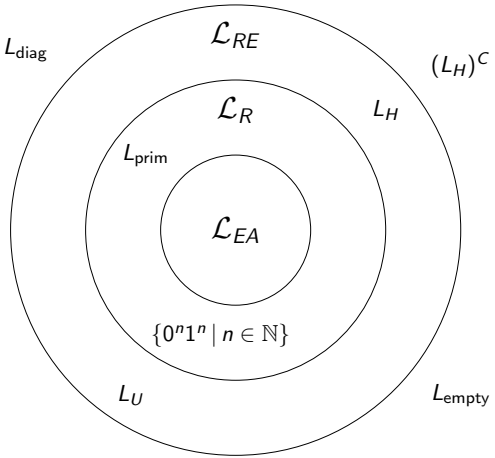


Figure: Courtesy of Nicolas Wehrli

Begrifflichkeiten

Für eine Sprache L gilt folgendes:

$$L \text{ regulär} \iff L \in \mathcal{L}_{\mathbf{EA}} \iff \exists \text{ EA } A \text{ mit } L(A) = L$$
$$L \text{ rekursiv} \iff L \in \mathcal{L}_{\mathbf{R}} \iff \exists \text{ Alg. A mit } L(A) = L$$
$$L \text{ rekursiv aufzählbar} \iff L \in \mathcal{L}_{\mathbf{RE}} \iff \exists \text{ TM } M \text{ mit } L(M) = L$$

- "Algorithmus" = TM, die immer hält
- L rekursiv = L entscheidbar
- L rekursiv aufzählbar = L erkennbar

Üben

Bait

- Blatt 6: Aufgabe 17 a)
- PL und Kolmogorov funktionieren nicht immer
- Kolmogorov: nur für $\Sigma = \{0, 1\}$
- PL: k=0 auch möglich
- Beweise
- lim-Notation korrekt verwenden

Aufgabe 14

(a) Entwerfen Sie einen nichtdeterministischen endlichen Automaten für die folgende Sprache in Diagrammdarstellung und erläutern Sie die Idee Ihres Entwurfs.

$$L_1 = \{w \in \{0,1\}^* \mid w = x0y0z \text{ mit } x, z \in \{0,1\}^* \text{ und } y \in \{0,1\}^2\}.$$

Figure: Blatt 5, 14a)

Üben

(b) Zeigen Sie, dass folgende Sprache nicht regulär ist:

$$L_2 = \{0^p \mid p \text{ ist eine Primzahl}\}$$

Figure: Blatt 5, 13b)

Üben

Verwenden Sie die Methode der Kolmogorov-Komplexität, um zu zeigen, dass die Sprache

$$L = \{w \in \{0,1\}^* \mid w = xxx \text{ für } x \in \{0,1\}^*\}$$

nicht regulär ist.

Figure: HS22 5a)

Üben

Zeigen Sie, dass jede unendliche Sprache eine nicht rekursiv aufzählbare Teilmenge enthält.

Figure: HS21 7

Üben

Beweise, dass die folgenden Sprachen nicht regulär sind. Verwende alle drei Methoden einmal:

- $\{0^{n^3} \mid n \in N\}$
- $\{x\#y \mid x, y \in \{0,1\}^*, |x|_1 = |y|_0\}$
- $\{0^n 1^{2n} 0^n \mid n \in N\}$

Üben

Zeichne einen NEA, welcher die Sprache $L = \{01, 001\}^*$ akzeptiert.

Üben

Zeigen Sie, dass zwei Konstanten $c, d \in \mathbb{N}$ existieren, so dass für alle $a, b \in \mathbb{N} - \{0, 1\}$ gilt, dass

$$K\left(0^{3a+7b}\right) \leq c \cdot (\log_2 a + \log_2 b) + d\,.$$

Figure: HS22 6

Übung

Zeichne einen EA, welcher die folgenden Sprachen akzeptiert. Beweise auch, dass dieser minimal ist.

$$\{x \in \Sigma_{bool}^* \mid |x| \bmod 4 = 2\}$$

Üben

Für jedes $n \in \mathbb{N} - \{0\}$ und jedes Wort $x = x_1 x_2 \dots x_n$ mit $x_1 \in \{1, 2, 3\}$ und $x_2, \dots, x_n \in \{0, 1, 2, 3\}$ bezeichne

$$\text{Nummer}_4(x) = \sum_{i=1}^{n} x_i \cdot 4^{n-i}$$

die natürliche Zahl mit der Darstellung x im Vierer-Zahlensystem.

Konstruieren Sie einen deterministischen endlichen Automaten in graphischer Darstellung für die Sprache

$$L_1 = \{w \in \{1, 2, 3\} \cdot \{0, 1, 2, 3\}^* \mid \text{Nummer}_4(w) \bmod 3 = 0\}$$

aller gültigen Vierer-Zahlendarstellungen (ohne führende Nullen) von durch 3 teilbaren Zahlen. Begründen Sie kurz die Korrektheit Ihrer Konstruktion.

Hinweis: Die Teilbarkeit durch 3 im Vierer-Zahlensystem lässt sich wie im Dezimalsystem durch Quersummenbildung überprüfen.

Figure: HS22 2a)