Vorlesung

Grundlagen der Theoretischen Informatik / Einführung in die Theoretische Informatik I

Bernhard Beckert

Institut für Informatik

UNIVERSITÄT KOBLENZ · LANDAU

Sommersemester 2007

Dank

Diese Vorlesungsmaterialien basieren ganz wesentlich auf den Folien zu den Vorlesungen von

Katrin Erk (gehalten an der Universität Koblenz-Landau)

Jürgen Dix (gehalten an der TU Clausthal)

Ihnen beiden gilt mein herzlicher Dank.

– Bernhard Beckert, April 2007

Teil V

1. Determinierte Turing-Maschinen (DTMs)
2. Varianten von Turing-Maschinen
3. Indeterminierte Turing-Maschinen (NTMs)
4. Universelle determinierte Turing-Maschinen
5. Entscheidbar/Aufzählbar
6. Determinierte Turing-Maschinen entsprechen Typ 0
7. **Unentscheidbarkeit**

Beispiel einer nicht berechenbaren Funktion: Busy Beaver

Definition 15.1 (Busy Beaver)

Die Funktion $BB : \mathbb{N} \to \mathbb{N}$ sei wie folgt definiert

$n \mapsto f(n) :=$ die maximale Anzahl an Einsen, die eine **haltende** DTM mit maximal n Zuständen auf einem leeren Band erzeugen kann

Beispiel einer nicht berechenbaren Funktion: Busy Beaver

BB wächst extrem schnell

Exakte Werte von $BB(n)$ für $n \geq 4$ nicht bekannt.

- $BB(4) \geq 4098$
- $BB(5) \geq 1,29 * 10^{865}$

Theorem 15.2 (BB ist nicht berechenbar)

BB wächst zu stark um berechenbar zu sein:
Es gibt keine DTM, die BB berechnet.

Beispiel einer nicht berechenbaren Funktion: Busy Beaver

Beweis (erster Teil)

Man kann immer mindestens ein $|$ mehr erzeugen, wenn man einen weiteren Zustand zur Verfügung hat:

- man benennt den Haltezustand um in q_{neu} und geht in den richtigen Haltezustand h nur, wenn man in q_{neu} ein Blank # gelesen hat. Zusätzlich ersetzt man das Blank durch $|$).
- Wenn man ein $|$ liest, geht man nach rechts und bleibt in q_{neu}.

Damit haben wir bewiesen:

***BB* wächst streng monoton.**

Beispiel einer nicht berechenbaren Funktion: Busy Beaver

Beweis (zweiter Teil)

Angenommen, es gäbe eine DTM $\mathcal{BB}$, die *BB* berechnet.
Sie habe n_0 Zustände.

Wir betrachten folgende zusammengesetzten Maschine:

- zuerst schreibt sie m Einsen auf das leere Band
- dann führt sie $\mathcal{BB}$ aus

Diese Maschine kommt mit $n_0 + \lfloor \frac{m}{2} \rfloor + 10$ Zuständen aus

Sei nun m so groß, dass

$$m > n_0 + \lfloor \frac{m}{2} \rfloor + 10$$

Dann schreibt die neue Maschine $BB(m)$ Einsen auf das Band und arbeitet mit streng weniger als m Zuständen: Widerspruch. □

Beispiel einer nicht berechenbaren Funktion: Busy Beaver

Intuitive Ursache des Widerspruchs

$\mathcal{BB}$ selbst hat fixe Größe,

Sie muss mehr Einsen schreiben können als jede Maschine ihrer Größe.

Also muss sie insbesondere mehr Einsen schreiben als sie selbst

Widerspruch

Göderlisierung von DTMs

Definition 15.3 (Göderlnummern von DTMs)

DTMn werden als Gödelzahlen kodiert:

- DTMs können als Gödelwörter dargestellt werden.
- Die Buchstaben der Gödelwörter können in Ziffern kodiert wrden, um Gödelnummern zu bekommen.
- Notation: $\hat{g}(\mathcal{M})$ für die Gödelnummer der DTM $\mathcal{M}$.

Göderlisierung von DTMs

Definition 15.4 (Jede Zahl ist Gödelnummer)

Jede natürliche Zahl n soll Gödelnummer einer DTM $\mathcal{M}_n$ sein.

Wir definieren

$$\mathcal{M}_n := \begin{cases} \mathcal{M}, & \text{falls } \hat{g}(\mathcal{M}) = n \\ \mathcal{M}_{halt} & \text{falls } \nexists \mathcal{M} \;\; \hat{g}(\mathcal{M}) = n \end{cases}$$

$\mathcal{M}_{halt}$ ist eine TM, die sofort anhält und weiter nichts tut.

Halteproblem

Definition 15.5 (Allgemeines Halteproblem)

Das **allgemeine Halteproblem** ist die Frage, ob die DTM mit Gödelnummer n bei Eingabe i hält.

Es entspricht der Sprache

$$\mathcal{H}_{allg} := \{\langle n, i\rangle \mid \mathcal{M}_n \text{ hält bei Eingabe } i\}.$$

Halteproblem

Definition 15.6 (Spezielles Halteproblem)

Das **spezielle Halteproblem** ist die Frage, ob die DTM mit Gödelnummer n bei Eingabe n hält.

Es entspricht der Sprache

$$\mathcal{H} := \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}.$$

Halteproblem

Definition 15.7 (Null-Halteproblem)

Das **Null-Halteproblem** ist die Frage, ob die DTM mit Gödelnummer n bei leerer Eingabe hält.

Es entspricht der Sprache

$$\mathcal{H}_0 := \{n \mid \mathcal{M}_n \text{ hält bei leerer Eingabe}\}$$

Manchmal auch:
"bei Eingabe 0" anstatt "bei leerer Eingabe".

Leerheitsproblem

Definition 15.8 (Leerheitsproblem)

Das **Leerheitsproblem** ist die Frage, ob die DTM mit Gödelnummer n bei keiner Eingabe aus Σ^* hält.

Es entspricht der Sprache

$$\mathcal{E} := \{n \mid \mathcal{M}_n \text{ hält bei keiner Eingabe aus } \Sigma^*\}.$$

Totalitätsproblem

Definition 15.9 (Totalitätsproblem)

Das **Totalitätsproblem** ist die Frage, ob die DTM mit Gödelnummer n bei jeder Eingabe aus Σ^* hält.

Es entspricht der Sprache

$$\mathcal{T} := \{n \mid \mathcal{M}_n \text{ hält bei jeder Eingabe aus } \Sigma^*\}.$$

Gleichheitsproblem

Definition 15.10 (Gleichheitsproblem)

Das **Gleichheitsproblem** ist die Frage, ob die DTM mit Gödelnummer n die gleiche Sprache über Σ akzeptiert wie die DTM mit Gödelnummer m.

Es entspricht der Sprache

$$\mathcal{E}q := \{\langle n, m\rangle \mid \mathcal{M}_n \text{ akzeptiert die gleiche Sprache über } \Sigma \text{ wie } \mathcal{M}_m\}.$$

Entscheidbarkeitsproblem

Definition 15.11 (Entscheidbarkeitsproblem)

Das **Entscheidbarkeitsproblem** ist die Frage, ob die DTM mit Gödelnummer n eine entscheidbare Sprache über Σ akzeptiert.

Es entspricht der Sprache

$$\mathcal{E}nt := \{n \mid \quad \mathcal{M}_n \text{ akzeptiert eine entscheidbare Sprache über } \Sigma\}.$$

Unentscheidbarkeit des Halteproblems

Satz 15.12 (Halteproblem ist unentscheidbar)

Das spezielle Halteproblem $\mathcal{H}$ ist unentscheidbar.

Unentscheidbarkeit des Halteproblems

Beweis (A. Turing)

Beweis durch Widerspruch mit einem **Diagonalisierungsargument**.

Angenommen, es gebe eine DTM $\mathcal{H}$,
die das spezielle Halteproblem entscheidet.

Konstruiere eine neue Maschine aus $\mathcal{H}$:

- Wenn $\mathcal{H}$ „Y“ antwortet,
  geht sie in eine Endlosschleife (termniniert nicht).
- Wenn $\mathcal{M}_n$ „N“ antwortet,
  terminiert sie.

Die neue Maschine habe Gödelnummer n.

Unentscheidbarkeit des Halteproblems

Beweis (A. Turing), Forts.

Was macht $\mathcal{M}_n$ bei Eingabe n?

- Falls $\mathcal{M}_n$ bei Eingabe n terminiert,
  dann antwortet $\mathcal{H}$ auf Eingabe n mit „N“,
  dann terminiert $\mathcal{M}_n$ auf Eingabe von n nicht
  Widerspruch!
- Falls $\mathcal{M}_n$ bei Eingabe n nicht terminiert,
  dann antwortet $\mathcal{H}$ auf Eingabe n mit „Y“,
  dann terminiert $\mathcal{M}_n$ auf Eingabe von n
  Widerspruch!

Akzeptierbarkeit des Halteproblems

Satz 15.13 (Akzeptierbarkeit von $\mathcal{H}$)

Das spezielle Halteproblem $\mathcal{H}$ ist aufzählbar. D. h. die Sprache

$$\mathcal{H} = \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}$$

ist akzeptierbar.

Beweis

Akzeptieren durch Simulation von $\mathcal{M}_n$ mit Hilfe der universellen DTM.

Korollar

Das Komplement von $\mathcal{H}$ ist nicht aufzählbar.

Reduktion von Problemen

Wie zeigt man, daß ein Problem unentscheidbar ist?

Reduktion (informell)

Wir geben eine **totale, berechenbare Funktion** f angeben, die

- eine Instanz p_1 von P_1
- in eine Instanz p_2 von P_2 umwandelt,
- und zwar so, daß die Antwort zu p_1 „ja" ist gdw die Antwort zu p_2 „ja" ist.

Reduktion von Problemen

Definition 15.14 (Reduktion)

Seien L_1, L_2 Sprachen über $\mathbb{N}$.

L_1 wird auf L_2 reduziert,

$$L_1 \preceq L_2$$

gdw

es gibt eine TM-berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$, so daß gilt:

$$\forall n \in \mathbb{N} \quad \big(n \in L_1 \text{ gdw } f(n) \in L_2\big).$$

Reduktion von Problemen

Lemma 15.15

*Ist $L_1 \preceq L_2$, und ist L_1 **unentscheidbar**, so ist auch L_2 **unentscheidbar**.*

Reduktion von Problemen

Beweis.

- Angenommen, L_2 ist entscheidbar.
- Sei $\mathcal{M}_2$ eine Turing-Maschine, die L_2 entscheidet.
- Wegen $L_1 \preceq L_2$ gibt es eine Funktion $f : \mathbb{N} \to \mathbb{N} \in TM$ mit $n \in L_1$ <u>gdw</u> $f(n) \in L_2$.
- Sei $\mathcal{M}_f$ eine DTM, die f berechnet.
- Dann kann man daraus die Maschine $\mathcal{M}_1 := \mathcal{M}_f\mathcal{M}_2$ konstruieren, für die gilt:
  - $\mathcal{M}_1$, gestartet mit Input n, hält mit $h, \#Y\underline{\#}$, falls $f(n) \in L_2$, d.h. wenn $n \in L_1$ ist.
  - $\mathcal{M}_1$, gestartet mit Input n, hält mit $h, \#N\underline{\#}$, falls $f(n) \notin L_2$, d.h. wenn $n \notin L_1$ ist.
- Die Maschine $\mathcal{M}_1$ entscheidet also L_1, ein Widerspruch.

□

Unentscheidbarkeit

Satz 15.16 (Unentscheidbarkeit von $\mathcal{H}_0$)

Das Null-Halteproblem

$$\mathcal{H}_0 = \{n \mid \mathcal{M}_n \textit{ hält bei Eingabe } 0\}$$

ist unentscheidbar.

Unentscheidbarkeit

Beweis

Gegeben eine TM $\mathcal{M}_n$.

Kombiniere diese mit einer DTM, die n aufs Band schreibt.

$f(n)$ sei definiert als die Gödelnummer dieser neuen Maschine.

$\mathcal{M}_{f(n)}$ terminiert auf Eingabe von 0
<u>gdw</u>
$\mathcal{M}_n$ terminiert auf Eingabe von n

Damit:
Reduktion des speziellen Halteproblems auf das Null-Halteproblem.
(f ist total und berechenbar!)

Also:
Unentscheidbarkeit des Null-Halteproblems folgt aus
Unentscheidbarkeit des speziellen Halteproblems

Unentscheidbarkeit

Weitere unentscheidbare Probleme

Ähnlich kann man per Reduktion die Unentscheidbarkeit der folgenden Probleme zeigen:

- $\mathcal{E}$, das Leerheitsproblem.
- $\mathcal{T}$, das Totalitätsproblem.
- $\mathcal{E}q$, das Gleichheitsproblem.
- $\mathcal{E}nt$, das Entscheidbarkeitsproblem.