Prof. Dr. Jasmin Blanchette
Elisabeth Lempa
Luca Maio

Ludwig-Maximilians-Universität München
Institut für Informatik
Besprechung 30.06.2025 bis 07.07.2025
Abgabe bis 14.07.2025, 10:00 Uhr

Lösungsvorschlag zur Übung 9 zur Vorlesung Theoretische Informatik für Studierende der Medieninformatik

TIMI9-1 ***Halteprobleme***

a) Betrachten Sie die Sprache

$$L_1 = \{w_M \in \{0,1\}^* \mid \text{TM } M \text{ hält für Eingabe } 01\}$$

wobei w_M das Encoding für M ist.

Welche der folgenden Aussagen ist korrekt? Beweisen Sie Ihre Antwort.

- L_1 ist entscheidbar.
- L_1 ist semi-entscheidbar, aber nicht entscheidbar.
- L_1 ist weder entscheidbar noch semi-entscheidbar.

Um zu zeigen, dass L_1 entscheidbar (bzw. semi-entscheidbar) ist, beschreiben Sie kurz die Funktionsweise einer deterministischen Turingmaschine, die L_1 (semi-) entscheidet. Um zu zeigen, dass L_1 nicht (semi-)entscheidbar ist, reduzieren Sie ein geeignetes Problem auf L_1.

LÖSUNGSVORSCHLAG:
L_1 ist semi-entscheidbar: Für eine Eingabe z simulieren wir die Maschine M_z mit Eingabe 01. Wenn die Simulation hält, geben wir 1 aus.

L_1 ist nicht entscheidbar. Wir beweisen das durch Reduktion von H_0 auf L_1.

Um $H_0 \leq L_1$ zu zeigen, müssen wir eine totale, berechenbare Funktion $f\colon \{0,1\}^* \to \{0,1\}^*$ angeben, sodass für alle z gilt: $z \in H_0 \iff f(z) \in L_1$. Wir wählen

$$f(z) = \begin{cases} w_{M'} & \text{falls } z \text{ ein valides Encoding für eine Turingmaschine } M \text{ ist.} \\ z & \text{sonst} \end{cases}$$

Dabei ist

- M' eine Turingmaschine, die zunächst ihre Eingabe löscht und dann M ausführt.

Die Funktion f ist offensichtlich total. Sie ist auch berechenbar, denn:

- Wir können Turingmaschinen binär kodieren und dekodieren (und erkennen, ob ein binäres Wort eine Turingmaschine kodiert).
- Wir können M' aus M konstruieren.

Für M' gilt:

- Wenn M mit Eingabe ε hält, dann hält M' mit jeder beliebigen Eingabe und also insbesondere mit Eingabe 01.
- M' hält mit einer beliebigen Eingabe, und also insbesondere mit Eingabe 01, nur dann, wenn M mit Eingabe ε hält.

Um die Äquivalenz $z \in H_0 \iff f(z) \in L_1$ zu zeigen, beweisen wir die beiden Implikationen $z \in H_0 \Rightarrow f(z) \in L_1$ und $z \notin H_0 \Rightarrow f(z) \notin L_1$.

$z \in H_0 \Rightarrow f(z) \in L_1$:

$$
\begin{array}{ll}
 & z \in H_0 \\
\Rightarrow & z \text{ ist ein valides Encoding für } M \text{ und } M \text{ hält mit Eingabe } \varepsilon \\
\Rightarrow & f(z) = w_{M'} \text{ und } M' \text{ hält mit jeder Eingabe (insbesondere 01)} \\
\Rightarrow & f(z) \in L_1
\end{array}
$$

$z \notin H_0 \Rightarrow f(z) \notin L_1$:

$$
\begin{array}{lll}
 & & z \notin H_0 \\
\text{Fall 1:} & & z \text{ ein valides Encoding für } M \\
 & \Rightarrow & M \text{ hält nicht mit Eingabe } \varepsilon \\
 & \Rightarrow & f(z) = w_{M'} \text{ und } M' \text{ hält auf keiner Eingabe (insbesondere nicht 01)} \\
 & \Rightarrow & f(z) \notin L_1 \\
\text{Fall 2:} & & z \text{ kein valides Encoding} \\
 & \Rightarrow & f(z) = z \text{ ist kein valides Encoding} \\
 & \Rightarrow & f(z) \notin L_1
\end{array}
$$

Somit ist $H_0 \leq L_1$ und da H_0 unentscheidbar ist, ist auch L_1 unentscheidbar.

b) Zeigen Sie, dass das folgende Problem für jede deterministische Turingmaschine M und natürliche Zahl n entscheidbar ist.

„M hält auf jeder Eingabe nach höchstens n Schritten.“

LÖSUNGSVORSCHLAG:
Für jede Eingabe w können wir prüfen, ob M auf w in n Schritten hält, indem wir M für n Schritte simulieren. Weiterhin kann M in n Schritten nur höchstens n Zeichen lesen. Deshalb gilt für Wörter w mit $|w| > n$: Wenn M

nicht in n Schritten auf $w[0] \dots w[n]$ hält, dann hält M auch nicht in n Schritten auf w. Somit genügt es, nur Wörter mit Länge höchstens n zu testen. Da das endlich viele Wörter sind, ist das Problem entscheidbar.

TIMI9-2 ***Entscheidbarkeit***

Prüfen Sie, ob die folgenden Behauptungen wahr oder falsch sind. Begründen Sie Ihre Antworten wie folgt: Wenn eine Sprache L (semi-)entscheidbar ist, beschreiben Sie die Funktionsweise einer deterministischen Turingmaschine, die die charakteristische Funktion χ_L bzw. χ'_L berechnet. Wenn L nicht (semi-)entscheidbar ist, leiten Sie einen Widerspruch ab.

a) Wenn A und B entscheidbare Sprachen sind, dann ist $A \cap B$ entscheidbar.

LÖSUNGSVORSCHLAG:
Wahr. Sei T_A eine DTM, die A entscheidet, und T_B eine DTM, die B entscheidet. Konstruiere eine DTM für $A \cap B$, die sich wie folgt verhält: Gegeben x, berechne $T_A(x)$. Falls $T_A(x) = 0$, lehne x ab, ansonsten berechne $T_B(x)$. Gilt $T_B(x) = 0$, lehne x ab, sonst akzeptiere x. Da T_A, T_B die jeweiligen Mengen entscheiden, terminieren beide DTM immer, womit auch die DTM zu $A \cap B$ stets mit dem korrekten Ergebnis terminiert. Somit ist $A \cap B$ entscheidbar.

b) Wenn A und $A \cup B$ entscheidbar sind, dann ist B entscheidbar.

LÖSUNGSVORSCHLAG:
Falsch. Sei $A = \{0,1\}^*$ und $B = H_0 \subseteq \{0,1\}^*$. Dann sind A und $A \cup B = A$ entscheidbar, aber B nicht.

c) Das Problem, ob $L(M) \neq \emptyset$ für eine gegebene Turingmaschine M gilt, ist semi-entscheidbar.

LÖSUNGSVORSCHLAG:
Wahr. Algorithmus: Für $i = 0, 1, \dots$ schreibe nacheinander alle Wörter $w \in \Sigma^*$ mit $|w| \leq i$ auf das Band und simuliere M auf jedem w für i Schritte. Falls M in i Schritten akzeptiert, gib 1 aus. Ist $L(M) \neq \emptyset$, so testen wir M irgendwann für ausreichend viele Schritte auf Wörtern ausreichender Länge, um ein Wort aus $L(M)$ zu finden.