Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe T10

Es sei $L(M) = \{ w \mid M \text{ akzeptiert } w \}$. Sind die folgenden Sprachen rekursiv, rekursiv aufzählbar oder keins von beidem. Beweisen Sie ihre Aussage.

1. $L_1 = \{ \langle M \rangle \mid L(M) = \emptyset \}$

   Lösungsvorschlag

   Nach dem Satz von Rice ist L_1 nicht rekursiv. Trivialerweise ist $M_\neg$, die TM die nichts akzeptiert, in der Sprache und M_A, die TM die alles akzeptiert, nicht in der Sprache.
   Außerdem ist die Sprache nicht rekursiv aufzählbar, da das Komplement rekursiv aufzählbar ist (vgl. folgende Teilaufgabe). Wären beide Sprachen rekursiv aufzählbar, dann wären auch beide Sprachen rekursiv.

2. $L_2 = \{ \langle M \rangle \mid L(M) \neq \emptyset \}$

   Lösungsvorschlag

   L_2 ist das Komplement von L_1, daher kann L_2 nicht rekursiv sein.
   Außerdem ist L_2 rekursiv aufzählbar. Hierzu kann ein Aufzähler konstruiert werden, der wie folgt arbeitet: Gehe aller möglichen Eingabewörter $w_i \in \{ 0, 1 \}^*$ und Turing Maschinen M_i in kanonischer Reihenfolge durch und simuliere M_i sowie alle Maschinen M_j mit $j < i$ auf w_i und allen Wörtern w_l mit $l < i$ für $|w|$ Schritte. Wird ein Wort w_l von einer TM M_j akzeptiert, dann gebe die Gödelnummer $\langle M_j \rangle$ aus.

3. $L_3 = \{ \langle M \rangle \mid |L(M)| \text{ ist endlich} \}$

   Lösungsvorschlag

   L_3 ist nach dem Satz von Rice nicht rekursiv. $\langle M_\neg \rangle \in L_3$ und $\langle M_A \rangle \notin L_3$.
   Außerdem ist L_3 auch nicht rekursiv aufzählbar. Hierzu reduzieren wir $\overline{H}_\epsilon$ auf L_3. Hierzu geben wir eine Funktion f an für die gilt: $w \in \overline{H}_\epsilon \Leftrightarrow f(w) \in L_3$. Im folgenden sei eine solche Funktion f konstruktiv als TM beschrieben:

   - Syntax-Check: Falls Eingabe w keine Gödelnummer ist, dann sei $f(w) = \langle M_\neg \rangle$ mit $M_\neg$ verwirft alle Wörter.
   - Falls $w = \langle M \rangle$ eine Gödelnumer ist, dann sei $f(w) = \langle M' \rangle$ mit M' löscht seine Eingabe und arbeitet auf ϵ. Falls M' hält, dann akzeptiert es immer.

Korrektheit:

$$\begin{aligned}
w \in \overline{H}_\epsilon &\Rightarrow w \neq \langle M \rangle \text{ oder } M \text{ hält nicht auf } \epsilon \\
&\Rightarrow M_\neg \text{ akzeptiert kein Wort oder } M \text{ hält nicht auf } \epsilon \\
&\Rightarrow L(M_neg) = \emptyset = L(M') \\
&\Rightarrow f(w) \in L_3 \\
w \notin \overline{H}_\epsilon &\Rightarrow w = \langle M \rangle \\
&\Rightarrow M \text{ hält auf } \epsilon \\
&\Rightarrow M' \text{ akzeptiert alles} \\
&\Rightarrow |L(M')| \text{ nicht endlich} \\
&\Rightarrow f(w) \notin L_3
\end{aligned}$$

4. $L_4 = \{\, \langle M \rangle \langle M' \rangle \mid L(M) \cap L(M') = \emptyset \,\}$

   Lösungsvorschlag

   L_7 ist nicht rekursiv. Sei M' die TM die alles akzeptiert, dann ist $L(M) \cap L(M') = \emptyset \Leftrightarrow L(M) = \emptyset$. Die Sprache die übrig bleibt $L' = \{\, \langle M \rangle \mid L(M) = \emptyset \,\}$ ist aber nach dem Satz von Rice nicht entscheidbar. Wieder ist $\langle M_\neg \rangle \in L'$ und $\langle M_A \rangle \notin L'$.
   Die Sprache ist nicht rekursiv aufzählbar, da wir ihr Komplement $\overline{L_6}$ rekursiv aufzählen können. Ein Aufzähler für $\overline{L_6}$ würde jeweils abwechselnd ein Wort aus $L(M)$ und $L(M')$ auf jeweils ein Band schreiben und bei jedem Wort überprüfen ob es schon auf dem anderen Band steht.

Aufgabe T11
Es seien $f, g \colon \mathbb{Z}^n \to \mathbb{Z}$ Polynome mit mehreren Variablen.

$$L_{f \leq g} = \{\, f, g \mid f \leq g \text{ hat eine ganzzahlige Lösung} \,\}$$

Ist $L_{f \leq g}$ entscheidbar? Beweisen sie ihre Aussage.
Hinweis: Eine Beispielinstanz wäre etwa $x + (x+7) \cdot y \leq z^7 - xy + 13$.

Lösungsvorschlag
Das Problem ist nicht entscheidbar. Um das zu zeigen nutzen wir eine Reduktion vom berühmten zehnte hilbertsche Problem, welches nach VL unentscheidbar ist. Dazu wandeln wir eine Instanz aus für das hilbertsche Problem in eine Instanz für $L_{f \leq g}$ um. Zu zeigen ist dann, dass die neue Instanz genau eine Lösung hat, wenn unsere original Instanz eine Lösung hat.
Wir setzen g auf die Nullfunktion ($g = 0$). Jetzt bleibt zu fragen ob $f \leq 0$ gilt. Sei f die Eingabe für das zehnte hilbertsche Problem. f hat genau dann eine ganzzahlige Nullstelle, wenn $f^2 \leq 0 = g$ eine ganzzahlige Lösung hat.

Aufgabe H10 (5 Punkte)
Ist das Problem aus Aufgabe T11 rekursiv aufzählbar? Beweisen Sie ihre Aussage.

Lösungsvorschlag

Ja es ist aufzählbar. Es gibt nur endlich viele verschiedene Variablenbelegungen. Diese werden nacheinander durchgegangen und sobald eine erfüllende dabei ist, geben wir diese aus.

Aufgabe H11 (5 Punkte)

Es seien $f, g : \{0,1\}^* \cup \{\perp\} \to \{0,1\}^* \cup \{\perp\}$ partielle Funktionen für die $f(\perp) = \perp$ und $g(\perp) = \perp$ gilt. Beweisen oder widerlegen sie:

Es gibt genau dann eine Turingmaschine, die $f \circ g$ berechnet, wenn es Turingmaschinen M und M' gibt, die f und g berechnen.

Lösungsvorschlag

$\Leftarrow$: Es werden einfach die beiden TMs nacheinander simuliert.

$\Rightarrow$: Die andere Richtung muss nicht unbedingt immer gelten. Sei z.B. f die Funktion die alles auf 0 abbildet, diese ist trivialerweise berechenbar. Nun sei g eine Funktion die das Halteproblem lösen möchte, diese können wir aus der TM die $f \perp g$ berechnet allerdings nicht herleiten.

Aufgabe H12 (8 Punkte)

Ist die Sprache $L = \{ \langle M \rangle \langle M' \rangle \mid L(M) \subseteq L(M') \}$ rekursiv aufzählbar? Beweisen Sie ihre Aussage.

Lösungsvorschlag

Die Sprache ist nicht rekursiv aufzählbar. Wir definieren die Funktion f, die die Codierung einer Turingmaschine $\langle M \rangle$ auf die Codierung zweier Maschinen $\langle M_a \rangle \langle M \rangle^*$ abbildet, wobei M_a eine feste Turingmaschine sei, die alle Eingaben akzeptiert und M^* eine Turingmaschine, die M simuliert und eine Eingabe w akzeptiert wenn M auf w hält.

Weil $L(M_a) = \{0,1\}^*$ ist, gilt folgendes:

$$\langle M \rangle \in H_{All} \Leftrightarrow f(\langle M \rangle) \in L_{H12}$$

Daher können wir mit einem hypothetischen Aufzähler $M_{\#L}$ für L einen Aufzähler $M_{\#H_{All}}$ konstruieren: wir gönnen uns eine Turingmaschine mit drei Bändern und simulieren $M_{\#L}$ auf dem ersten Band. Die Ausgabe von $M_{\#L}$ geben wir dabei auf das zweite Band, dort erscheinen also nacheinander alle Wörter aus L. Gleichzeit (d.h. im Wechselschritt mit der Simulation von $M_{\#L}$) zählen wir auf dem dritten Band Turingmaschinen auf und schreiben für jede Maschine M die Codierung $\langle M \rangle \# f(M) \#\#$, also $\langle M \rangle \# \langle M_a \rangle \langle M^* \rangle \#\#$ auf dieses Band. Nun wird unser simulierter Aufzähler irgendwann den String $\langle M_a \rangle \langle M^* \rangle$ ausgeben, sollte M tatsächlich auf allein Eingaben halten. Passiert dies, so löschen wir $\langle M_a \rangle \langle M^* \rangle$ vom zweiten und $\langle M \rangle \# \langle M_a \rangle \langle M^* \rangle \#\#$ vom dritten Band und geben M aus.

Diese Konstruktion wird jede Maschine M, die auf allen Eingaben hält, nach endlicher Zeit ausgeben und ist damit ein Aufzähler für H_{All}. Wir schließen daraus, dass es keinen Aufzähler für L geben kann.