8. Aufgabenblatt

(Besprechung in den Tutorien 11.12.2022–15.12.2022)

Aufgabe 1. Totalitätsproblem (Fortsetzung)

Sei Σ ein endliches Alphabet. Eine Sprache $L \subseteq \Sigma^*$ heißt *co-semi-entscheidbar*, falls $\overline{L}$ semi-entscheidbar ist. Wie im 7. Aufgabenblatt definieren wir das *Totalitätsproblem* durch:

$$T = \{w \in \{0,1\}^* \mid M_w \text{ hält bei jeder möglichen Eingabe } x \in \{0,1\}^*\}.$$

(a) Seien $A, B \subseteq \Sigma^*$. Zeigen Sie, dass, wenn $A \leq B$ und B co-semi-entscheidbar ist, dann auch A co-semi-entscheidbar ist.

(b) Schlussfolgern Sie aus (a), dass T nicht co-semi-entscheidbar ist.

(c) Zeigen Sie, dass T nicht semi-entscheidbar ist.

(d) Schlussfolgern Sie aus (c), dass nicht $T \leq K$ gilt, wobei K das spezielle Halteproblem ist.

Anmerkung: Dies zeigt, dass T in einem gewissen Sinne „echt schwerer“ ist als alle semi-entscheidbaren und alle co-semi-entscheidbaren Sprachen.

——————Lösungsskizze——————

(a) Wie in der Vorlesung gezeigt, gilt $\overline{A} \leq \overline{B}$. Da B co-semi-entscheidbar ist, ist $\overline{B}$ semi-entscheidbar. Nach Vorlesung folgt, dass $\overline{A}$ semi-entscheidbar ist. Daraus folgt wiederum, dass A co-semi-entscheidbar ist.

(b) Im 7. Aufgabenblatt haben wir gesehen, dass $K \leq T$, wobei K das spezielle Halteproblem ist. Aus der Vorlesung ist bekannt, dass K semi-entscheidbar, aber nicht entscheidbar ist. Daraus folgt, dass K nicht co-semi-entscheidbar ist. Nach (a) folgt dann wiederum, dass T nicht co-semi-entscheidbar ist.

(c) Wir zeigen, dass $\overline{K} \leq T$. Da $\overline{K}$ nicht semi-entscheidbar ist, folgt, dass auch T nicht semi-entscheidbar ist.

Wir geben nun eine Reduktion von $\overline{K}$ auf T an. Sei $w \in \{0,1\}^*$ und M_w die durch w kodierte Turing-Maschine. Wir konstruieren eine Turing-Maschine $\tau(M_w)$, die auf Eingabe $x = x_0 \dots x_k \in \{0,1\}^*$ Folgendes tut:

Sei $n := \sum_{i=0}^{k} x_i \cdot 2^i$ die Zahl, die durch x binär kodiert wird (für $x = \varepsilon$ sei $n := 0$). Die TM $\tau(M_w)$ schreibt zunächst w aufs Band und führt dann n Schritte von M_w auf Eingabe w aus. Falls M_w innerhalb dieser n Schritte halten würde, so geht $\tau(M_w)$ in eine Endlosschleife. Falls M_w innerhalb dieser n Schritte nicht gehalten hat, so hält $\tau(M_w)$. Die Reduktionsfunktion ist $f\colon \{0,1\}^* \to \{0,1\}^*$ mit $f(w) := \langle \tau(M_w) \rangle$. Diese Funktion ist total und berechenbar.

Es bleibt zu zeigen, dass $w \in \overline{K} \iff f(w) \in T$.

Falls $w \in \overline{K}$, so hält die TM M_w nicht auf Eingabe w. Das heißt, es gibt kein $n \in \mathbb{N}$, sodass M_w innerhalb von n Schritten auf Eingabe w hält. Folglich hält $\tau(M_w)$ auf allen Eingaben $x \in \{0,1\}^*$. Somit gilt $f(w) \in T$.

Falls $f(w) \in T$, so hält $\tau(M_w)$ auf allen Eingaben $x \in \{0,1\}^*$. Folglich gibt es kein $n \in \mathbb{N}$, sodass M_w auf Eingabe w innerhalb von n Schritten hält. Das heißt, dass M_w gar nicht auf Eingabe w hält. Also gilt $w \in \overline{K}$.

(d) Da K semi-entscheidbar ist, aber T nicht, kann $T \leq K$ nicht gelten.

Aufgabe 2. (Semi-)Entscheidbarkeit
Ist die Sprache

$$L := \{w \in \{0,1\}^* \mid M_w \text{ akzeptiert ein Wort der Länge } 1\}$$

semi-entscheidbar? Ist sie entscheidbar?

Hinweis: Sie können die Existenz einer *universellen* Turing-Maschine, die eine beliebige andere Turing-Maschine "simulieren" kann, annehmen.

——————Lösungsskizze——————

L ist semi-entscheidbar. Sei Σ das Eingabealphabet von M_w. Eine (universelle) TM, die L semi-entscheidet, simuliert M_w für alle Eingaben $a \in \Sigma$ "parallel" (d.h. abwechselnd mit zunehmender Schrittzahl).

L ist unentscheidbar. Wir benutzen den Satz von Rice in der Formulierung über akzeptierte Sprachen. Sei $\mathcal{A}$ die Menge aller Sprachen vom Typ 0. Sei $\mathcal{S} = \{A \in \mathcal{A} \mid A \text{ enthält ein Wort der Länge } 1\}$. Es gilt $\mathcal{S} \neq \emptyset$ und $\mathcal{S} \neq \mathcal{A}$. Folglich ist $\mathcal{C}(\mathcal{S}) = \{w \mid T(M_w) \in \mathcal{S}\} = L$ nach dem Satz von Rice unentscheidbar.

Aufgabe 3. Satz von Rice
Ziegen Sie für jede der folgenden Sprachen entweder, dass diese unentscheidbar ist (mittels Satz von Rice), oder, dass sie entscheidbar ist.

(a) $\{w \in \{0,1\}^* \mid |T(M_w)| = 12\}$

(b) $\{w \in \{0,1\}^* \mid M_w \text{ besitzt eine gerade Anzahl von Zuständen}\}$

(c) $\{w \in \{0,1\}^* \mid |T(M_w)| = \infty\}$

(d) $\{w \in \{0,1\}^* \mid \varepsilon \in T(M_w)\}$

(e) $\{w \in \{0,1\}^* \mid T(M_w) \text{ enthält mindestens ein Wort ungerader Länge}\}$

——————Lösungsskizze——————

Sei $\mathcal{A}$ die Menge aller Typ-0 Sprachen.

(a) Wähle $\mathcal{S} = \{A \in \mathcal{A} \mid |A| = 12\}$.

(b) Entscheidbar: Wir können uns die Kodierung der Turing-Maschine anschauen und ihre Zustände einfach zählen.

(c) Wähle $\mathcal{S} = \{A \in \mathcal{A} \mid |A| = \infty\}$.

(d) Wähle $\mathcal{S} = \{A \in \mathcal{A} \mid \epsilon \in A\}$.

(e) Wähle $\mathcal{S} = \{A \in \mathcal{A} \mid A \text{ enthält ein Wort ungerade Länge}\}$.

Aufgabe 4. Fleißige Biber

Ein *unärer fleißiger Biber* ist eine Turing-Maschine

$$B = (\{z_0, \ldots, z_n\}, \{1\}, \{1, \Box\}, \delta, z_0, \Box, \{z_n\})$$

mit n Nicht-Endzuständen, die bei Eingabe des leeren Wortes in endlich vielen Schritten hält und dabei die maximal mögliche Anzahl 1'en aufs Band schreibt, verglichen mit allen anderen Turing-Maschinen, welche die gleichen Voraussetzungen erfüllen (n Nicht-Endzustände, Alphabete $\{1\}$ und $\{1, \Box\}$ und Halten bei leerer Eingabe).

a) Geben Sie eine Überführungsfunktion δ an, sodass die Turing-Maschine

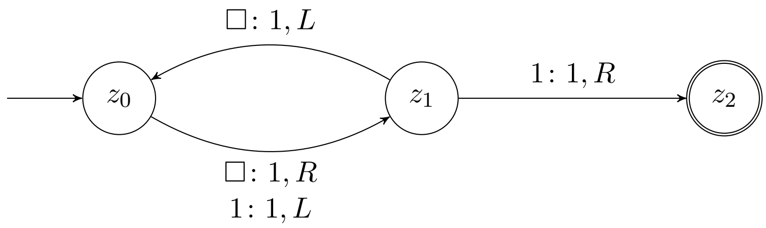
$$(\{z_0, z_1, z_2\}, \{1\}, \{1, \Box\}, \delta, z_0, \Box, \{z_2\})$$

bei leerer Eingabe möglichst viele 1'en aufs Band schreibt und hält.

b) Ist die Sprache $\{w \in \{0,1\}^* \mid M_w \text{ ist ein fleißiger Biber}\}$ entscheidbar?
Sie können davon ausgehen, dass die folgende Funktion unberechenbar ist: $b : \mathbb{N} \to \mathbb{N}$, $b(n)$ ist die Anzahl 1'en, die ein fleißiger Biber mit n Nicht-Endzuständen aufs Band schreibt. Außerdem können Sie die Existenz einer *universellen* Turing-Maschine, die eine beliebige andere Turing-Maschine "simulieren" kann, annehmen.

——————Lösungsskizze——————

a) $B_2 = (\{z_0, z_1, z_2\}, \{1\}, \{1, \Box\}, \delta, z_0, \Box, \{z_2\})$, wobei δ wie folgt definiert ist:



Bei dem leeren Wort als Eingabe ergibt sich die folgende Konfigurationsfolge:

$$z_0\Box \vdash^1_{B_2} 1z_1\Box \vdash^1_{B_2} z_0 11 \vdash^1_{B_2} z_1\Box 11 \vdash^1_{B_2} z_0\Box 111 \vdash^1_{B_2} 1z_1 111 \vdash^1_{B_2} 11z_2 11.$$

b) Die Sprache ist nicht entscheidbar, denn wenn sie entscheidbar wäre, könnte man die Funktion b wie folgt berechnen:

Bei Eingabe n, iteriere über jede mögliche Kodierung $\langle M \rangle$ einer Turing-Maschine M mit n Nicht-Endzuständen (endlich viele) und überprüfe, ob diese in der Sprache liegt. Wenn M ein fleißiger Biber ist (Beachte: Es existiert immer einer), dann simuliere M auf der leeren Eingabe, um $b(n)$ 1'en auf dem Band zu erhalten.