Prof. T. Nipkow
M. Eberl, J. Rädle, S. Wimmer
Technische Universität München
KEINE ABGABE

Einführung in die theoretische Informatik

Sommersemester 2019 – Übungsblatt Lösungsskizze 10

AUFGABE 10.1. (*Wichtige Begriffe*) Stufe A

Überprüfen Sie, dass Sie die Folgenden Begriffe korrekt definieren können.

- Reduktion
- Satz von Rice
- PCP

AUFGABE 10.2. (*Falsche Reduktionen*) Stufe B

Sei $\mathsf{H}_0 := \{w \mid \exists \mathsf{M}.\mathsf{enc}(\mathsf{M}) = w \wedge \mathsf{M}[\varepsilon]\downarrow\}$ und sei $A := \{w \in \Sigma^* \mid \exists i \in \mathbb{N}_0.\ |w| = 5i + 3\}$ mit $\Sigma = \{\mathsf{a}, \mathsf{b}\}$. Erklären Sie, warum die angegebenen Funktionen keine Reduktionen gemäß Vorlesungsdefinition sind.

(a) Behauptung: $\mathsf{H}_0 \leq A$
Reduktion:
$$f(w) = \begin{cases} \mathsf{aaa} & \text{falls } w \in \mathsf{H}_0 \\ \mathsf{b} & \text{sonst} \end{cases}$$

(b) Behauptung: $A \leq \mathsf{H}_0$
Reduktion: f bildet jedes Element $x \in \Sigma^*$ auf die Kodierung einer TM M_x, die wie folgt definiert ist: Die TM M_x löscht die Eingabe und schreibt x aufs Band, bestimmt dann die Länge von x, zieht 3 ab und prüft anschließend, ob das Ergebnis durch 5 teilbar ist. Dementsprechend gibt die Maschine "Ja" (1) und "Nein" (0) aus.

(c) Behauptung: $\overline{\mathsf{H}_0} \leq \mathsf{H}_0$
Reduktion: f bildet jedes $w \in \{0,1\}^*$ auf die Kodierung $f(w)$ einer TM $\mathsf{M}_{f(w)}$ ab, die $\mathsf{M}_w[\varepsilon]$ simuliert. Falls M_w hält, geht $\mathsf{M}_{f(w)}$ in eine Endlosschleife. Falls $\mathsf{M}_w[\varepsilon]$ nicht hält, hält $\mathsf{M}_{f(w)}$.

(d) Behauptung: $H_{\Sigma^*} \leq \mathsf{H}_0$ mit $H_{\Sigma^*} = \{w \in \{0,1\}^* \mid \exists \mathsf{M}.\mathsf{enc}(\mathsf{M}) = w \wedge \forall x \in \Sigma^*.\ \mathsf{M}[x]\downarrow\}$.
Reduktion: f bildet jedes $w \in \{0,1\}^*$ auf die Kodierung $f(w)$ einer TM $\mathsf{M}_{f(w)}$ ab, die erst die Eingabe löscht und nicht deterministisch $x \in \Sigma^*$ erzeugt und dann $\mathsf{M}_w[x]$ simuliert.

Lösungsskizze

(a) f ist unberechenbar, da H_0 unentscheidbar ist und somit χ_{H_0} unberechenbar ist.

(b) f bildet auf Kodierungen von Turing-Maschinen ab, die immer terminieren. Da $\mathsf{a} \notin A$, aber $f(\mathsf{a}) \in \mathsf{H}_0$, erfüllt die Funktion f nicht die Definition einer Reduktion.
Außerdem ist die Notation M_x ungünstig, da wir einen Index einer TM in der Regel verwenden, um anzuzeigen, dass M_w die TM ist, die von w encodiert wird, also $\mathsf{enc}(\mathsf{M}) = w$. In dieser Reduktion hat M_x aber eine andere Bedeutung.

(c) f ist nicht wohldefiniert. Wenn $\mathsf{M}_{f(w)}$ die Berechnung von $\mathsf{M}_w[\varepsilon]$ simuliert und $\mathsf{M}_w[\varepsilon]$ nicht hält, dann hält definitiv $\mathsf{M}_{f(w)}$ auch nicht.

(d) Sei M irgendeine Turing-Maschine mit $\mathsf{M}[\varepsilon]\downarrow$ und $\neg\mathsf{M}[0]\downarrow$ und sei $w \in \Sigma^*$ mit $\mathsf{M}_w = \mathsf{M}$. Dann gilt $w \notin H_{\Sigma^*}$ und $f(w) \in \mathsf{H}_0$.

AUFGABE 10.3. (*Satz von Rice*) Stufe C

Entscheiden Sie, ob die folgenden Mengen unentscheidbar für $\Sigma = \{0,1\}$ sind, und begründen Sie Ihre Antworten mit dem Satz von Rice (falls anwendbar). Geben Sie dabei die Menge $\mathcal{F}$ genau an und argumentieren Sie, warum die Menge nicht trivial ist.

(a) $\mathsf{L}_1 = \{w \in \Sigma^* \mid \{u \in \Sigma^* \mid \varphi_w(u) = 1\}\ \mathsf{ist\ regulär}\}$
(b) $\mathsf{L}_2 = \{w \in \Sigma^* \mid \forall n \in \mathbb{N}_0.\ \varphi_w(n) = n * (n - 23) + 42\}$
(c) $\mathsf{L}_3 = \{w \in \Sigma^* \mid \forall p \in \mathbb{N}_0.\ (|w| > p \wedge p \text{ ist prim}) \rightarrow w_p = 0\}$

Hinweis: $w_p \in \Sigma$ bezeichnet den Buchstaben an der p-ten Stelle im Wort w.

Lösungsskizze

(a) Sei $\mathcal{F} = \{f \mid f \text{ is berechenbar } \wedge f^{-1}(1) \text{ ist regulär}\}$. Sei nun $g(w) = 1$ und

$$h(w) = \begin{cases} 1 & \text{falls } \exists i \geq 0.\ w = 0^i 1^i \\ 0 & \text{sonst} \end{cases}$$

zwei berechenbare Funktionen. Dann gilt $g \in \mathcal{F}$ und $h \notin \mathcal{F}$. Somit ist $\mathcal{F}$ nicht die Menge aller berechenbarer Funktionen. Damit folgt aus dem Satz von Rice, dass L_1 unentscheidbar ist.

(b) Sei $\mathcal{F} = \{f \mid f \text{ ist berechenbar } \wedge \forall n \in \mathbb{N}_0.\ f(n) = n * (n-23) + 42\}$. Dann gilt für $g(n) = 0$: $g \notin \mathcal{F}$ und somit ist $\mathcal{F}$ nicht die Menge aller berechenbarer Funktionen. Weiterhin ist $\mathcal{F}$ auch nicht leer, da das Polynom in der Definition berechenbar ist. Somit ist nach Satz von Rice L_2 unentscheidbar.

(c) L_3 ist entscheidbar, da w nur syntaktischen Kriterien erfüllen muss. Eine TM kann alle Primzahlen kleiner $|w|$ berechnen und an diesen Stellen in w prüfen, ob $w_p = 0$ gilt.

AUFGABE 10.4. (*Reduktionen*) Stufe C

Betrachten Sie die folgende Menge:

$$A \coloneqq \{w \in \{0,1\}^* \mid \exists \mathsf{M}.w = \mathsf{enc}(\mathsf{M}) \wedge \exists x \in \{0,1\}^*.\ \varphi_w(x) = |x|\},$$

wobei φ_w die von M berechnete Funktion ist.
Zeigen Sie die folgenden Behauptungen:

(a) A ist nicht entscheidbar.
  (i) Geben Sie eine passende Reduktion von H_0 an und begründen Sie deren Korrektheit. Verwenden Sie $\mathsf{H}_0 := \{w \mid \exists \mathsf{M}.w = \mathsf{enc}(\mathsf{M}) \wedge \mathsf{M}[\varepsilon]\downarrow\}$.
  (ii) Verwenden Sie den Satz von Rice.

(b) A ist semi-entscheidbar.

(c) $\overline{A}$ ist nicht semi-entscheidbar.

Lösungsskizze

(a) (i) $\mathsf{H}_0 \leq A$:
Reduktion: f bildet jedes $w \in \{0,1\}^*$, sodass $w = \mathsf{enc}(\mathsf{M})$ auf die Kodierung einer TM M' ab, die erst die Eingabe löscht und dann $\mathsf{M}[\varepsilon]$ simuliert. Sobald die simulierte Turing-Maschine hält, gibt M' 0 aus. Worte, die kein TM encodieren, werden auf ein beliebiges Element $y \notin A$ abgebildet. So ein y existiert, z.B. die Maschine, die immer nach rechts läuft und nie hält.

$$f(w) = \begin{cases} \mathsf{enc}(\mathsf{M}') & \text{if } w = \mathsf{enc}(\mathsf{M}) \\ y & \text{else} \end{cases}$$

Berechenbarkeit: Diese Reduktion ist berechenbar, da das Löschen der Eingabe, sowie die Simulation berechenbar sind.
Korrektheit:
- Sei $w \in \mathsf{H}_0$, dann $\mathsf{M}'[x]\downarrow$ für alle $x \in \{0,1\}^*$ mit Ausgabe 0, also insbesondere für $x = \varepsilon$. Somit $f(w) \in A$.
- Sei $w \notin \mathsf{H}_0$ und $\mathsf{M} = \mathsf{dec}(w)$, dann hält $\mathsf{M}'[x]$ nie für alle $x \in \{0,1\}^*$ und somit $f(w) \notin A$.
- Sei $w \notin \mathsf{H}_0$ und $\nexists \mathsf{M}.\mathsf{M} = \mathsf{dec}(w)$. Dann ist auch $f(w) = y \notin A$

Da H_0 unentscheidbar ist, ist A auch unentscheidbar.

(ii) Zu entscheiden, ob eine TM mindestens für eine Eingabe die Länge der Eingabe ausgibt, bedeutet, eine nicht-triviale semantische Eigenschaft der TM zu entscheiden. Das ist laut dem Satz von Rice nicht möglich.
Formaler: $\mathcal{F} = \{f \mid f \text{ ist berechenbar} \wedge \exists x : f(x) = |x|\}$. Die Menge ist nicht-trivial, da z.B. $g(x) = |x|$ ein Element von Ihr ist, aber $h(x) = |x| + 1$ nicht. Folglich ist die Menge nicht entscheidbar.

(b) **Algorithmus:** Die TM enumeriert für $i = 0, 1, 2, \ldots$ alle Eingaben $x \in \bigcup_{k=0}^{i} \{0,1\}^k$ für die TM M_w, simuliert $\mathsf{M}_w[x]$ für i Schritte und prüft, ob $\varphi_w(x) = |x|$. Falls dieser Test wahr ist, terminiert die Maschine und akzeptiert w.
Korrektheit: Sei $w \in A$. Dann terminiert der Algorithmus, da es eine Eingabe x mit $\varphi_w(x) = |x|$ gibt und die $\mathsf{M}_w[x]$ nach einer fixen Anzahl an Schritten terminiert. Sei $w \notin A$. Dann terminiert der Algorithmus nie, da die Abbruchbedingung nie wahr ist.
Alternativ: Eine NTM kann das Wort raten, bei dem die Länge des Wortes berechnet wird und dann die Berechnung durchführen. Existiert kein solches Wort, terminiert die Maschine nicht.

(c) Anwendung von Satz 5.41 liefert das gewünschte Ergebnis, da A unentscheidbar, aber semi-entscheidbar ist. Alternativ: Aus (a) folgt (nach Übungsaufgabe 9.4(b)) $\overline{\mathsf{H}_0} \leq \overline{A}$. Da $\overline{\mathsf{H}_0}$ nicht semi-entscheidbar ist, ist $\overline{A}$ nicht semi-entscheidbar.

AUFGABE 10.5. (*PCP*)
Wir betrachten in dieser Aufgabe das Post'sche Korrespondenzproblem (PCP).

(a) Bestimmen Sie *alle* Lösungen für das folgende PCP: $P_1 = ((d, cd), (d, d), (abc, ab))$.
(b) Zeigen Sie, dass die folgende Instanz des PCPs keine Lösung hat: $P_2 = ((ab, aba), (baa, aa), (aba, baa))$.
(c) Zeigen Sie, dass das Post'sche Korrespondenzproblem über einem Alphabet mit nur einem Symbol entscheidbar ist, indem Sie einen Algorithmus angeben. Begründen Sie auch dessen Korrektheit.
(d) Sei $P = (c_1, c_2)$ ein PCP über einem beliebigem Alphabet Σ mit $c_i = (x_i, y_i)$ und $||x_i| - |y_i|| = 1$ für $i \in \{1, 2\}$. Zeigen Sie die Entscheidbarkeit für diese Variante des PCPs. Geben Sie hierzu einen Algorithmus an und begründen Sie dessen Korrektheit.

Lösungsskizze

(a) Die Menge aller Lösungen ist: $\mathsf{L}((2 \mid (31))^*) \setminus \{\varepsilon\}$
(b) Zu Beginn kann nur die Karte (ab, aba) verwendet werden, da 2 und 3 als Startkarte ungeeignet sind und somit alle Lösungen mit 1 beginnen müssen. Wir müssen deshalb mit dem Überhang (ε, a) fortfahren. Offensichtlich kann nun weder (ab, aba) (Karte 1) noch (baa, aa) (Karte 2) angewendet werden. Durch (aba, baa) (Karte 3) erhalten wir aber $(aba, abaa)$, was wiederum zu dem Überhang (ε, a) führt. Wir können somit keinen Abschluss finden und damit kann diese Instanz des Post'schen Korrespondenzproblems keine Lösung besitzen.
(c) In diesem Fall haben alle Karten c die Form $c = (a^i, a^j)$ für $\Sigma = \{a\}$ und $i, j \geq 0$.
- Falls $i = j$, ist die Karte c alleine eine Lösung.
- Wenn alle Karten oben länger als unten ($i > j$) sind, gibt es keine Lösung.
- Analog für den umgekehrten Fall ($i < j$).
- Wenn es zwei Karten c_1, c_2 mit $c_1 = (a^{j_1}, a^{k_1})$, $c_2 = (a^{j_2}, a^{k_2})$, $j_1 > k_1$ und $j_2 < k_2$ gibt, sei i_1 der Index von c_1 und sei i_2 der von c_2. Dann ist z.B. $i_1^{k_2 - j_2} i_2^{j_1 - k_1}$ eine Lösung des PCP.
  Darauf kann man kommen, indem man eine Lösung für das folgende Gleichungssystem sucht:

$$\begin{aligned} j_1 \cdot \#_{c1} + j_2 \cdot \#_{c2} &= k_1 \cdot \#_{c1} + k_2 \cdot \#_{c2} \\ j_1 &> k_1 \\ j_2 &< k_2 \end{aligned}$$

Eine TM kann diese Vorbedingungen prüfen und somit bestimmen, ob es eine Lösung gibt.

(d) Entscheidbar, da es genau dann eine Lösung gibt, wenn 12 oder 21 eine Lösung ist, und diese beiden Fälle von einer TM geprüft werden können.
Beweis:
Sei $i_1 i_2 \dots i_k$ eine kürzeste Lösung von P. Aus der Längenbedingung der Karten folgt sofort, dass $k \geq 2$ gilt. Falls $i_1 \neq i_2$, dann gilt $|x_{i_1} x_{i_2}| = |y_{i_1} y_{i_2}|$. Somit ist bereits $i_1 i_2$ (12 oder 21) eine Lösung.
Sei nun $i_1 = i_2$ und o.B.d.A. $i_1 = i_2 = 1$. Wir nehmen ebenfalls o.B.d.A. an, dass $|x_1| > |y_1|$ gilt. Somit $x_1 = u_1 \dots u_n u_{n+1}$ und $y_1 = u_1 \dots u_n$ mit $u_i \in \Sigma$ gilt. Da $i_1 i_2$ Teil einer Lösung ist, gilt für $(x_1 x_1, y_1 y_1)$:

$$x_1 x_1 = y_1 y_1 u_n u_{n+1}$$

und somit

$$u_{n+1} = u_1 = u_2 = \dots = u_{n-1} = u_n$$

Die Karte c_1 hat somit die Gestalt $c_1 = (a^{n+1}, a^n)$ für irgendeinen Buchstaben $a \in \Sigma$.
In der Lösung muss es auch eine solche Teilsequenz (22) für c_2 geben, damit der Überhang von c_1 ausgeglichen wird. Ähnlich wie für c_1 folgt dann $c_2 = (b^m, b^{m+1})$ für ein $b \in \Sigma$. Da c_1 und c_2 sich überschneiden, gilt $a = b$. Somit ist auch 12 eine Lösung.

AUFGABE 10.6. (*Entscheidbarkeit und kontextfreie Grammatiken*) Stufe D
Seien $\mathsf{G}_1, \mathsf{G}_2$ CFGs. Beweisen Sie die folgenden beiden Aussagen:

(a) $\mathsf{L}(\mathsf{G}_1) \not\subseteq \mathsf{L}(\mathsf{G}_2)$ ist semi-entscheidbar.
(b) $\mathsf{L}(\mathsf{G}_1) \subseteq \mathsf{L}(\mathsf{G}_2)$ ist unentscheidbar.
Hinweis: Betrachten Sie den Beweis für die Unentscheidbarkeit von $\mathsf{L}(\mathsf{G}_1) \cap \mathsf{L}(\mathsf{G}_2) = \emptyset$ aus der Vorlesung.

Lösungsskizze

(a) Es gilt $\mathsf{L}(\mathsf{G}_1) \not\subseteq \mathsf{L}(\mathsf{G}_2)$ genau dann, wenn es ein $w \in \mathsf{L}(\mathsf{G}_1) \setminus \mathsf{L}(\mathsf{G}_2)$ gibt.
Für $w \in \Sigma^*$: Für jedes w testet man mittels CYK (o.B.d.A. sind G_1 und G_2 in CNF), ob $w \in \mathsf{L}(\mathsf{G}_1)$ und ob $w \in \mathsf{L}(\mathsf{G}_2)$ gilt. Sobald man das erste $w \in \mathsf{L}(\mathsf{G}_1) \setminus \mathsf{L}(\mathsf{G}_2)$ gefunden hat, stoppt man und gibt 1 aus. Offensichtlich stoppt der Algorithmus im Fall $\mathsf{L}(\mathsf{G}_1) \not\subseteq \mathsf{L}(\mathsf{G}_2)$ stets, im Fall $\mathsf{L}(\mathsf{G}_1) \subseteq \mathsf{L}(\mathsf{G}_2)$ terminiert der Algorithmus allerdings nie. Damit ist das Problem semi-entscheidbar.

(b) Es gilt $\mathsf{L}(\mathsf{G}_1) \subseteq \mathsf{L}(\mathsf{G}_2)$ **gdw** $\mathsf{L}(\mathsf{G}_1) \setminus \mathsf{L}(\mathsf{G}_2) = \emptyset$ **gdw** $\mathsf{L}(\mathsf{G}_1) \cap \overline{\mathsf{L}(\mathsf{G}_2)} = \emptyset$. (O.B.d.A. verwenden G_1 und G_2 dasselbe Alphabet Σ.)
Sei $(x_1, y_1), \ldots, (x_l, y_l) \in \Gamma^* \times \Gamma^*$ eine PCP-Instanz. Sei $\Sigma = \{a_1, \ldots, a_l\}$, o.B.d.A. $\Gamma \cap \Sigma = \emptyset$ und $\mathsf{A} = \Sigma \cup \Gamma$, da wir Σ frei wählen können. Dann sind die in der Vorlesung verwendeten Grammatik $\mathsf{G}_1, \mathsf{G}_2$ deterministisch. Da DCFL unter Komplement nach Vorlesung abgeschlossen sind, kann man aus G_2 bzw. dem entsprechenden DPDA eine CFG G_2' mit $\mathsf{L}(\mathsf{G}_2') = \overline{\mathsf{L}(\mathsf{G}_2)}$ konstruieren.
Damit gilt: $\mathsf{L}(\mathsf{G}_1) \subseteq \mathsf{L}(\mathsf{G}_2')$ **gdw** $\mathsf{L}(\mathsf{G}_1) \cap \overline{\mathsf{L}(\mathsf{G}_2')} = \emptyset$ **gdw** $\mathsf{L}(\mathsf{G}_1) \cap \mathsf{L}(\mathsf{G}_2) = \emptyset$ **gdw** die gegebene PCP-Instanz hat keine Lösung. Somit ist die Teilmengenrelation über CFGs unentscheidbar.