

Lösungsvorschlag zur Übung 10 zur Vorlesung
 Theoretische Informatik für Studierende der Medieninformatik

TIMI10-1 PCP-Varianten

- a) Wir betrachten das PCP4-Problem, eine Variante von PCP, bei der die Spielsteine aus vier Wörtern bestehen. Eine Instanz von PCP4 ist also eine endliche Folge von 4-Tupeln $(x_1, y_1, z_1, u_1), \dots, (x_n, y_n, z_n, u_n)$ mit $x_i, y_i, z_i, u_i \in \Sigma^+$ für $i = 1, \dots, n$. Eine Lösung der Instanz K ist ähnlich zu PCP eine endliche Folge von Indices $i_1, \dots, i_m \in \mathbb{N}$, sodass $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m} = z_{i_1} \cdots z_{i_m} = u_{i_1} \cdots u_{i_m}$. Sind die folgenden Instanzen K_1, K_2 von PCP4 lösbar? Wenn ja, geben Sie eine Lösung (also eine geeignete Folge von Indizes) an. Wenn nein, beweisen Sie, dass die Instanz keine Lösung hat.

$$K_1 = \left(\begin{bmatrix} bbc \\ bca \\ cab \\ bbca \end{bmatrix}, \begin{bmatrix} a \\ ab \\ abb \\ a \end{bmatrix}, \begin{bmatrix} cca \\ ca \\ a \\ ca \end{bmatrix}, \begin{bmatrix} abc \\ bcc \\ ccc \\ bcc \end{bmatrix} \right)$$

$$K_2 = \left(\begin{bmatrix} c \\ b \\ bab \\ cc \end{bmatrix}, \begin{bmatrix} acb \\ baa \\ aaa \\ acba \end{bmatrix}, \begin{bmatrix} bca \\ caaa \\ b \\ c \end{bmatrix}, \begin{bmatrix} b \\ ba \\ b \\ bab \end{bmatrix} \right)$$

LÖSUNGSVORSCHLAG:

K_1 hat die Lösung 2, 1, 4, 3, also

$$\begin{bmatrix} a \\ ab \\ abb \\ a \end{bmatrix}, \begin{bmatrix} bbc \\ bca \\ cab \\ bbca \end{bmatrix}, \begin{bmatrix} abc \\ bcc \\ ccc \\ bcc \end{bmatrix}, \begin{bmatrix} cca \\ ca \\ a \\ ca \end{bmatrix}$$

K_2 hat keine Lösung: Mit den Spielsteinen 1, 2 und 3 kann man nicht beginnen, da alle mindestens ein Wort haben das bereits im ersten Symbol nicht mit den anderen übereinstimmt. Betrachten wir also den Fall, dass wir mit

dem Spielstein $\begin{bmatrix} b \\ ba \\ b \\ bab \end{bmatrix}$ anfangen. Wenn wir mit $\begin{bmatrix} c \\ b \\ bab \\ cc \end{bmatrix}$ oder $\begin{bmatrix} acb \\ baa \\ aaa \\ acba \end{bmatrix}$ weiter machen wollen, passen die ersten beiden Wörter nicht zusammen, da bc kein Präfix von bab bzw. ba kein Präfix von $baaa$ ist, also können diese keine Weiterführungen sein. Mit $\begin{bmatrix} bca \\ caaa \\ b \\ c \end{bmatrix}$ haben wir allerdings in den letzten beiden Wörtern einen Unterschied, da bb kein Präfix von bab ist. Somit gibt es keine mögliche Weiterführung und da $\begin{bmatrix} b \\ ba \\ b \\ bab \end{bmatrix}$ auch alleine noch keine Lösung ist, kann K_2 keine Lösung haben.

- b) Zeigen Sie durch Reduktion von PCP auf PCP4, dass PCP4 unentscheidbar ist.

LÖSUNGSVORSCHLAG:

Wir zeigen $\text{PCP} \leq \text{PCP4}$. Da PCP unentscheidbar ist, folgt daraus, dass auch PCP4 unentscheidbar ist.

Sei $K = ((x_1, y_1), \dots, (x_n, y_n))$ eine Instanz von PCP mit Alphabet Σ . Wir definieren $f(K) = f((x_1, y_1), \dots, f((x_n, y_n))$ und $f((x_i, y_i)) = (x_i, y_i, x_i, x_i)$. Damit ist $f(K)$ auf jeden Fall eine Instanz von PCP4 und f ist offensichtlich total und berechenbar.

Es gilt somit:

- Wenn $K \in \text{PCP}$ ist, dann hat K eine Lösung $i_1, \dots, i_m$ mit $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$. Da $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m} = x_{i_1} \cdots x_{i_m} = x_{i_1} \cdots x_{i_m}$, ist $i_1, \dots, i_m$ auch eine Lösung für $f(K)$, somit $f(K) \in \text{PCP4}$.
- Wenn $f(K) \in \text{PCP4}$ ist, dann hat $f(K)$ eine Lösung $i_1, \dots, i_m$ mit $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m} = x_{i_1} \cdots x_{i_m} = x_{i_1} \cdots x_{i_m}$. Somit gilt auch bereits $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$ und $i_1, \dots, i_m$ ist auch eine Lösung für K und $K \in \text{PCP}$.

- c) Wir betrachten das 456PCP-Problem, eine Variante von PCP, bei der die 'Spielsteine' auf das Alphabet $\Sigma = \{4, 5, 6\}$ beschränkt sind. Eine Instanz von 456PCP ist also eine endliche Folge von Paaren $(x_1, y_1), \dots, (x_n, y_n)$ mit $x_i, y_i \in \{4, 5, 6\}^+$ für $i = 1, \dots, n$. Eine Lösung der Instanz K ist wie bei PCP eine endliche Folge von Indices $i_1, \dots, i_m \in \mathbb{N}$, sodass $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$.

Zeigen Sie durch Reduktion von PCP auf 456PCP, dass 456PCP unentscheidbar ist.

LÖSUNGSVORSCHLAG:

Wir zeigen $PCP \leq 456PCP$. Da PCP unentscheidbar ist, ist damit auch 456PCP unentscheidbar.

Sei $K = ((x_1, y_1), \dots, (x_n, y_n))$ eine Instanz von PCP mit Alphabet $\Sigma = \{a_1, \dots, a_j\}$. Wir definieren $f(a_i) = 45^i$ für die Elemente aus Σ und $f(\varepsilon) = \varepsilon$, $f(a_i w) = f(a_i) f(w)$ für Wörter aus Σ^* , sowie schließlich $f(K) = (f(x_1), f(y_1)), \dots, (f(x_n), f(y_n))$ für Instanzen. Da 45^i ein Wort aus $\{4, 5, 6\}^+$ darstellt, überführt f Instanzen von PCP in Instanzen von 456PCP. Damit ist f auch offensichtlich total und berechenbar.

Weiter gilt:

- Wenn $K \in PCP$ ist, dann hat K eine Lösung $i_1, \dots, i_m$ mit $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$. Dann ist

$$f(x_{i_1}) \cdots f(x_{i_m}) = f(x_{i_1} \cdots x_{i_m}) = f(y_{i_1} \cdots y_{i_m}) = f(y_{i_1}) \cdots f(y_{i_m})$$

und es ist also $i_1, \dots, i_m$ eine Lösung für $f(K)$, somit $f(K) \in 456PCP$.

- Wenn $f(K) \in 456PCP$ ist, dann hat $f(K)$ eine Lösung $i_1, \dots, i_m$ mit $f(x_{i_1}) \cdots f(x_{i_m}) = f(y_{i_1}) \cdots f(y_{i_m})$. Die Zuordnung 45^i zu Buchstaben a_i ist eindeutig für ein gegebenes Alphabet Σ , damit können wir eine Funktion g angeben die sich invers zu f verhält. Es gilt also

$$\begin{aligned} x_{i_1} \cdots x_{i_m} &= g(f(x_{i_1} \cdots x_{i_m})) = g(f(x_{i_1}) \cdots f(x_{i_m})) = \\ &= g(f(y_{i_1}) \cdots f(y_{i_m})) = g(f(y_{i_1} \cdots y_{i_m})) = y_{i_1} \cdots y_{i_m} \end{aligned}$$

Somit ist $i_1, \dots, i_m$ auch eine Lösung für $K \in PCP$.

- d) Für Mengen Σ und Δ nennen wir eine Funktion $f : \Sigma^* \rightarrow \Delta^*$ einen *Homomorphismus* (siehe auch Aufgabe FSK6-4), wenn gilt:

$$\begin{aligned} f(\varepsilon) &= \varepsilon \\ f(u \circ v) &= f(u) \circ f(v) \quad \forall u, v \in \Sigma^* \end{aligned}$$

Wir definieren das Problem HOMPCP. Eine Instanz dieses Problem ist ein 4-Tupel (Σ, Δ, f, g) , wobei Σ und Δ endliche Mengen sind und $f, g : \Sigma^* \rightarrow \Delta^*$ Homomorphismen. Eine Lösung der Instanz ist ein Wort $w \in \Sigma^+$ sodass gilt: $f(w) = g(w)$.

Wir betrachten außerdem das LPCP-Problem, eine Variante von PCP, bei der die ‚Spielsteine‘ auch das leere Wort enthalten können. Eine Instanz von LPCP mit Alphabet Σ ist also eine endliche Folge von Paaren $(x_1, y_1), \dots, (x_n, y_n)$ mit $x_i, y_i \in \Sigma^*$.

Σ^* für $i = 1, \dots, n$ (wohingegen bei PCP gilt: $x_i, y_i \in \Sigma^+$). Eine Lösung der Instanz K ist wie bei PCP eine endliche Folge von Indices $i_1, \dots, i_m \in \mathbb{N}$ sodass $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$. Das LPCP-Problem ist unentscheidbar.

Zeigen Sie durch Reduktion von LPCP auf HOMPCP, dass HOMPCP für $|\Delta| \geq 2$ unentscheidbar ist.

LÖSUNGSVORSCHLAG:

Wir zeigen $\text{LPCP} \leq \text{HOMPCP}$. Da LPCP unentscheidbar ist, folgt daraus, dass auch HOMPCP unentscheidbar ist.

Sei $K = ((x_1, y_1), \dots, (x_n, y_n))$ eine Instanz von LPCP mit Alphabet Γ . Wir definieren eine Instanz $F(K) = (\Sigma, \Delta, f, g)$ von HOMPCP wie folgt:

$$\begin{aligned}\Sigma &= \{1, \dots, n\} \\ \Delta &= \Gamma \\ f(i) &= \begin{cases} \varepsilon & \text{für } i = \varepsilon \\ x_i & \text{für } i \in \Sigma \\ f(j) \circ f(k) & \text{für } i = j \circ k; j, k \in \Sigma^+ \end{cases} \\ g(i) &= \begin{cases} \varepsilon & \text{für } i = \varepsilon \\ y_i & \text{für } i \in \Sigma \\ g(j) \circ g(k) & \text{für } i = j \circ k; j, k \in \Sigma^+ \end{cases}\end{aligned}$$

Wörter aus Σ^+ sind dabei endliche, nicht leere Folgen $i_1 \cdots i_m$ von Indizes aus der Indexmenge Σ . Offensichtlich sind f und g per Definition Homomorphismen.

Nun gilt:

- Eine Lösung für K ist eine endliche, nicht leere Folge von Indizes $i_1, \dots, i_m \in \Sigma$. Sei $w = i_1 \cdots i_m$. Dann gilt:

$$f(w) = f(i_1) \cdots f(i_m) = x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m} = g(i_1) \cdots g(i_m) = g(w)$$

Somit ist w eine Lösung von $F(K)$.

- Umgekehrt ist eine Lösung von $F(K)$ ein nicht leeres Wort $w = i_1 \cdots i_m$ sodass

$$x_{i_1} \cdots x_{i_m} = f(i_1) \cdots f(i_m) = f(w) = g(w) = g(i_1) \cdots g(i_m) = y_{i_1} \cdots y_{i_m}$$

Somit ist $i_1, \dots, i_m$ eine Lösung von K .

F ist außerdem total und berechenbar und somit eine valide Reduktionsfunktion.

TIMI10-2 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 $f: \{0,1\}^* \rightarrow \{0,1\}^*$ 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 $z \notin H_0 \Rightarrow f(z) \notin L_1$.

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

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

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

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

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.