Informatik IV
Theoretische Informatik

Kapitel 11
Unentscheidbarkeit

Sommersemester 2019

Dozent: Prof. Dr. J. Rothe

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

Eigenschaft partiell rekursiver Funktionen

Ziel: Ein einfaches Kriterium für die Unentscheidbarkeit von Problemen.

Definition

Sei $G = (\varphi_0, \varphi_1, \varphi_2, \ldots)$ eine fixierte Gödelisierung von $\mathbb{P}$.

- Jede Teilmenge $F \subseteq \mathbb{P}$ ist eine *Eigenschaft partiell rekursiver Funktionen*.
- Die *Nummernmenge* (oder *Indexmenge*) von $F \subseteq \mathbb{P}$ bzgl. G ist

$$N_F = \{i \in \mathbb{N} \mid \varphi_i \in F\}.$$

- Eine Eigenschaft $F \subseteq \mathbb{P}$ heißt *nichttrivial*, falls $N_F \neq \emptyset$ und $N_F \neq \mathbb{N}$.

Der Satz von Rice

Theorem (Satz von Rice)

Die Nummernmenge einer jeden nichttrivialen Eigenschaft partiell rekursiver Funktionen ist unentscheidbar.
Das heißt: Für jedes nichttriviale $F \subseteq \mathbb{P}$ ist $N_F \notin \mathrm{REC}$.

Alternative Formulierung: Ist $\mathcal{A} \subseteq \mathrm{RE}$ eine nichttriviale Teilmenge von RE (d.h., $\emptyset \neq \mathcal{A} \neq \mathrm{RE}$), dann ist das folgende Problem unentscheidbar: Gegeben eine Turingmaschine M (als Wort, das M syntaktisch beschreibt), ist $L(M) \in \mathcal{A}$?

Interpretation: Aus der Syntax von Programmen kann nichts Nichttriviales über ihre Semantik algorithmisch geschlossen werden!

Beweis des Satzes von Rice

Beweis: Sei $F \subseteq \mathbb{P}$ eine nichttriviale Eigenschaft.

Für einen Widerspruch nehmen wir an, die Nummernmenge N_F wäre in REC.

Da $\emptyset \neq N_F \neq \mathbb{N}$, können wir feste Zahlen $a \in N_F$ und $b \notin N_F$ wählen.

Definiere die zweistellige Funktion

$$\alpha(i, x) = \begin{cases} \varphi_b(x) & \text{falls } i \in N_F \\ \varphi_a(x) & \text{falls } i \notin N_F. \end{cases}$$

Beweis des Satzes von Rice

Dann ist $\alpha \in \mathbb{P}$ mit dem folgenden Algorithmus. Für gegebene i und x:

- entscheide, ob
  - $i \in N_F$ oder
  - $i \notin N_F$

  (das geht nach Annahme $N_F \in \mathrm{REC}$);
- berechne die Programme der Turingmaschinen M_a und M_b;
- simuliere $M_b(x)$, falls $i \in N_F$;
- simuliere $M_a(x)$, falls $i \notin N_F$.

Beweis des Satzes von Rice

Da $\alpha \in \mathbb{P}$, existiert ein $m \in \mathbb{N}$ mit

$$\begin{aligned} \alpha(i,x) &= \varphi_m(i,x) \\ &= \varphi_{s(m,i)}(x) && \text{nach dem Iterationssatz, } s \in \mathbb{R} \\ &= \varphi_{g(i)}(x) && \text{da } m \text{ fest, } g \in \mathbb{R} \end{aligned}$$

Nach dem Kleeneschen Fixpunktsatz existiert ein $j \in \mathbb{N}$ mit

$$\varphi_j = \varphi_{g(j)}.$$

Daraus folgt für alle $x \in \mathbb{N}$:

$$\varphi_j(x) = \varphi_{g(j)}(x) = \alpha(j,x) = \begin{cases} \varphi_b(x) & \text{falls } j \in N_F \\ \varphi_a(x) & \text{falls } j \notin N_F. \end{cases}$$

Beweis des Satzes von Rice

Damit erhalten wir den folgenden Widerspruch:

- Ist $\varphi_j \in F$, so ist $j \in N_F$, also gilt $\varphi_j = \varphi_b$, woraus $\varphi_j \notin F$ mit $b \notin N_F$ folgt.
- Ist $\varphi_j \notin F$, so ist $j \notin N_F$, also gilt $\varphi_j = \varphi_a$, woraus $\varphi_j \in F$ mit $a \in N_F$ folgt.

Zusammengefasst gilt also:

$$\varphi_j \in F \iff \varphi_j \notin F,$$

ein Widerspruch. Die Annahme war falsch, und es gilt $N_F \notin \mathrm{REC}$. ❑

Anwendung des Satzes von Rice

Nachweis der Unentscheidbarkeit von Problemen, z.B.:

- $A = \{x \mid \varphi_x \text{ ist total}\} \notin \mathrm{REC}$.
  Denn es gibt *totale* ebenso wie *nicht-totale* partiell rekursive Funktionen. Also ist A die Nummernmenge einer nichttrivialen Eigenschaft von $\mathbb{P}$ und nach dem Satz von Rice nicht entscheidbar.

- $B = \{x \mid \|W_x\| = \infty\} \notin \mathrm{REC}$.
  Denn es gibt partiell rekursive Funktionen mit *endlichem*, aber ebenso welche mit *unendlichem* Wertebereich. Also ist B die Nummernmenge einer nichttrivialen Eigenschaft von $\mathbb{P}$ und nach dem Satz von Rice nicht entscheidbar.

Anwendung des Satzes von Rice

- $C = \{(x, y) \mid y \in \mathsf{W}_x\} \notin \text{REC}.$
- $D = \{(x, y) \mid \varphi_x \equiv \varphi_y\} \notin \text{REC}.$
- $E = \{(x, y, z) \mid \varphi_x(y) = z\} \notin \text{REC}.$
- $K = \{x \mid x \in \mathsf{D}_x\} \notin \text{REC}.$
- $\overline{K} = \{x \mid x \notin \mathsf{D}_x\} \notin \text{REC}.$
- $X = \{i \in \mathbb{N} \mid \mathsf{D}_i \neq \emptyset\} \notin \text{REC}.$
- $Y = \{x \mid 17 \in \mathsf{W}_x\} \notin \text{REC}.$
- ...

Reduzierbarkeit

Ziel: Durch (berechenbare) Reduktionen wollen wir ungelöste Probleme auf bereits gelöste Probleme zurückführen.
Damit erhalten wir ein weiteres Unentscheidbarkeitskriterium.

Definition

Seien $A, B \subseteq \mathbb{N}$ Mengen. A ist *many-one-reduzierbar* auf B (symbolisch: $A \leq_{\mathrm{m}} B$), falls gilt:

$$(\exists f \in \mathbb{R})\,(\forall x \in \mathbb{N})\,[x \in A \iff f(x) \in B].$$

Zur $\leq_{\mathrm{m}}$-Reduzierbarkeit äquivalente Aussagen

Lemma

Die folgenden Aussagen sind paarweise äquivalent:

1. $A \leq_{\mathrm{m}} B$ *via* $f \in \mathbb{R}$.
2. $A = f^{-1}(B)$, *wobei* $f^{-1}(B) = \{x \in \mathbb{N} \mid f(x) \in B\}$.
3. $f(A) \subseteq B$ *und* $f(\overline{A}) \subseteq \overline{B}$, *wobei* $f(A) = \{f(x) \in \mathbb{N} \mid x \in A\}$.
4. $\chi_A = \chi_B \circ f$.

ohne Beweis

Eigenschaften der $\leq_{\mathrm{m}}$-Reduzierbarkeit

Lemma

Seien $A, B \subseteq \mathbb{N}$ Mengen.

1. *Die Relation $\leq_{\mathrm{m}}$ ist reflexiv und transitiv, aber nicht antisymmetrisch.*
2. *Gilt $A \leq_{\mathrm{m}} B$ und ist B entscheidbar, so ist auch A entscheidbar.*
3. *Gilt $A \leq_{\mathrm{m}} B$ und ist B semi-entscheidbar, so ist auch A semi-entscheidbar.*
4. *Im Allgemeinen gilt nicht: $\overline{A} \leq_{\mathrm{m}} A$.*
5. *$A \leq_{\mathrm{m}} B \implies \overline{A} \leq_{\mathrm{m}} \overline{B}$.* **ohne Beweis**

Unentscheidbarkeitskriterium

Die Aussage:

2. Gilt $A \leq_{\mathrm{m}} B$ und ist B entscheidbar, so ist auch A entscheidbar.

aus dem vorigen Lemma sagt:

Entscheidbarkeit vererbt sich bzgl. der $\leq_{\mathrm{m}}$-Reduzierbarkeit nach unten.

In der Kontraposition verwenden wir diese Eigenschaft als:

Lemma (Unentscheidbarkeitskriterium)

Unentscheidbarkeit vererbt sich bzgl. der $\leq_{\mathrm{m}}$-Reduzierbarkeit nach oben:

$$(A \leq_{\mathrm{m}} B \wedge A \notin \mathrm{REC}) \implies B \notin \mathrm{REC}.$$

Das allgemeine Halteproblem

Definition

Die Sprache

$$H = \{w\#x \in \{0,1,\#\}^* \mid M_w(x) \text{ hält nach endlich vielen Schritten}\}$$

wird als das *allgemeine Halteproblem* bezeichnet.
Alternative Definition:

$$H = \{(i,j) \in \mathbb{N}^2 \mid i \in \mathsf{D}_j\}$$

Theorem

Das Halteproblem ist nicht entscheidbar, d.h., $H \notin \mathrm{REC}$.

Anwendung des Unentscheidbarkeitskriteriums

Beweis: Wenn wir zeigen, dass $K \leq_{\mathrm{m}} H$ gilt, folgt die Aussage des Satzes aus $K \notin \mathrm{REC}$ und dem Unentscheidbarkeitskriterium.

Die Funktion $f : \{0,1\}^* \to \{0,1,\#\}^*$ mit

$$f(w) = w\#w$$

ist total und berechenbar. Es gilt für alle $w \in \{0,1\}^*$:

$$\begin{aligned} w \in K \quad &\Leftrightarrow \quad M_w \text{ angesetzt auf } w \text{ hält} && \text{(nach Def. von } K\text{)} \\ &\Leftrightarrow \quad w\#w \in H && \text{(nach Def. von } H\text{)} \\ &\Leftrightarrow \quad f(w) \in H && \text{(nach Def. von } f\text{)} \end{aligned}$$

Damit gilt $w \in K \Leftrightarrow f(w) \in H$, also $K \leq_{\mathrm{m}} H$. ❑

Alternativer Beweis von $H \notin \mathrm{REC}$

Definiere die Reduktion

$$f(i) = (i, i).$$

Dann gilt:

$$\begin{aligned} i \in K \quad &\iff \quad i \in \mathsf{D}_i \\ &\iff \quad (i, i) = f(i) \in H. \end{aligned}$$

Da offenbar $f \in \mathbb{R}$, ist $K \leq_{\mathrm{m}} H$ via f.

Mit $K \notin \mathrm{REC}$ und dem Unentscheidbarkeitskriterium folgt $H \notin \mathrm{REC}$. ❑

Halteproblem auf leerem Band

Definition

Die Sprache

$$H_0 = \{w \mid M_w(\lambda) \text{ hält nach endlich vielen Schritten}\}$$

wird als das *Halteproblem auf leerem Band* bezeichnet.

Theorem

Das Halteproblem auf leerem Band ist nicht entscheidbar, d.h.,

$$H_0 \notin \mathrm{REC}.$$

Halteproblem auf leerem Band

Beweis: Es genügt zu zeigen, dass $H \leq_{\mathrm{m}} H_0$ gilt.

Wir ordnen jedem Wort z aus $\{0, 1, \#\}^*$ wie folgt eine deterministische Turingmaschine M zu.

- Falls z nicht von der Form $w\#x$ mit $w, x \in \{0, 1\}^*$ ist, so geht M in eine Endlosschleife.
- Falls z von der Form $w\#x$ mit $w, x \in \{0, 1\}^*$ ist, dann schreibt M das Wort x auf das Band und arbeitet dann, angesetzt auf x, wie M_w.

Halteproblem auf leerem Band

Wir definieren die berechenbare totale Funktion $f : \{0, 1, \#\}^* \to \{0, 1\}^*$ durch

$$f(z) = \mathrm{code}(M).$$

Es gilt für alle $z \in \{0, 1, \#\}^*$:

$$\begin{aligned}
& z \in H \\
\Leftrightarrow\quad & z = w\#x \text{ und } M_w \text{ angesetzt auf } x \text{ hält} && \text{(Def. von } H\text{)} \\
\Leftrightarrow\quad & f(z) = \mathrm{code}(M) \text{ und } M \text{ angesetzt auf } \lambda \text{ hält} && \text{(Def. von } M \text{ und } f\text{)} \\
\Leftrightarrow\quad & f(z) = \mathrm{code}(M) \in H_0 && \text{(Def. von } H_0\text{)}
\end{aligned}$$

Damit gilt $z \in H \Leftrightarrow f(z) \in H_0$. ❑

Anwendung des Unentscheidbarkeitskriteriums

Behauptung: Das Problem

$$X = \{i \in \mathbb{N} \mid \mathsf{D}_i \neq \emptyset\}$$

ist nicht entscheidbar.

Beweis: Wir zeigen: $K \leq_{\mathrm{m}} X$. Da $K \notin \mathrm{REC}$, folgt aus dem Unentscheidbarkeitskriterium $X \notin \mathrm{REC}$.

Sei die Funktion $\alpha : \mathbb{N}^2 \to \mathbb{N}$ definiert durch

$$\alpha(i, x) = \begin{cases} 1 & \text{falls } i \in K \\ \text{nicht definiert} & \text{sonst.} \end{cases}$$

Anwendung des Unentscheidbarkeitskriteriums

Da $\alpha \in \mathbb{P}$, gibt es ein $j \in \mathbb{N}$ mit $\varphi_j = \alpha$.

Nach dem Iterationssatz existiert eine Funktion $s \in \mathbb{R}$, so dass gilt:

$$\alpha(i, x) = \varphi_j(i, x) = \varphi_{s(j,i)}(x) = \varphi_{g(i)}(x),$$

wobei die letzte Gleichheit für ein geeignetes $g \in \mathbb{R}$ gilt, da j fest ist.

Es folgt:

$$\mathsf{D}_{g(i)} = \begin{cases} \mathbb{N} & \text{falls } i \in K \\ \emptyset & \text{falls } i \notin K. \end{cases}$$

Somit gilt für alle $i \in \mathbb{N}$:

$$\mathsf{D}_{g(i)} = \mathbb{N} \iff \mathsf{D}_{g(i)} \neq \emptyset.$$

Anwendung des Unentscheidbarkeitskriteriums

Also gilt für alle $i \in \mathbb{N}$:

$$\begin{aligned} i \in K \quad &\Longleftrightarrow \quad \mathsf{D}_{g(i)} \neq \emptyset \\ &\Longleftrightarrow \quad g(i) \in X. \end{aligned}$$

Folglich ist $K \leq_{\mathrm{m}} X$ via g. ❑

Bemerkung: Wir haben gezeigt:

$$K \leq_{\mathrm{m}} H \quad \text{und} \quad K \leq_{\mathrm{m}} X.$$

Ebenso gilt:

$$H \leq_{\mathrm{m}} K \quad \text{und} \quad X \leq_{\mathrm{m}} K.$$

Anwendung des Unentscheidbarkeitskriteriums

Wir zeigen die erste Behauptung:

$$H \leq_{\mathrm{m}} K.$$

Sei die Funktion $\alpha : \mathbb{N}^3 \to \mathbb{N}$ definiert durch

$$\alpha(i,j,x) = \begin{cases} 1 & \text{falls } (i,j) \in H \\ \text{nicht definiert} & \text{sonst.} \end{cases}$$

Da $\alpha \in \mathbb{P}$, existiert nach dem Iterationssatz eine Funktion $h \in \mathbb{R}$, so dass gilt:

$$\alpha(i,j,x) = \varphi_{h(i,j)}(x).$$

Anwendung des Unentscheidbarkeitskriteriums

Also gilt für alle $(i, j) \in \mathbb{N}^2$:

$$
\begin{aligned}
(i,j) \in H \quad &\Longleftrightarrow \quad \varphi_{h(i,j)}(x) \text{ ist für alle } x \text{ definiert} \\
&\Longleftrightarrow \quad \varphi_{h(i,j)}(h(i,j)) \text{ ist definiert} \\
&\Longleftrightarrow \quad h(i,j) \in \mathrm{D}_{h(i,j)} \\
&\Longleftrightarrow \quad h(i,j) \in K.
\end{aligned}
$$

Folglich ist $H \leq_{\mathrm{m}} K$ via h. ❑

$\leq_{\mathrm{m}}$-vollständige Mengen in RE

Bemerkung: K, H und X sind also alle „$\leq_{\mathrm{m}}$-äquivalente“ Probleme, denn

$$K \leq_{\mathrm{m}} H, \quad K \leq_{\mathrm{m}} X, \quad H \leq_{\mathrm{m}} K \quad \text{und} \quad X \leq_{\mathrm{m}} K.$$

Ebenso sind sie alle rekursiv aufzählbar.

Sie sind Beispiele für „$\leq_{\mathrm{m}}$-vollständige“ Probleme in RE.

Das heißt, *jede* Menge in RE lässt sich auf ein solches vollständiges Problem reduzieren.

$\leq_{\mathrm{m}}$-vollständige Mengen in RE

Definition

Eine Menge A heißt *$\leq_{\mathrm{m}}$-vollständig in RE*, falls gilt:

1. $A \in \mathrm{RE}$ und
2. $(\forall B \in \mathrm{RE})\,[B \leq_{\mathrm{m}} A]$.

Theorem

Die Probleme K, H und X sind $\leq_{\mathrm{m}}$-vollständig in RE. **ohne Beweis**

$\leq_{\mathrm{m}}$-vollständige Mengen in RE

Theorem

Alle Mengen, die $\leq_{\mathrm{m}}$-vollständig in RE sind, sind unentscheidbar.

Beweis: Sei A ein $\leq_{\mathrm{m}}$-vollständiges Problem in RE.

Das heißt, jedes Problem aus RE lässt sich auf A reduzieren.

Insbesondere gilt $K \leq_{\mathrm{m}} A$.

Da das Halteproblem K unentscheidbar ist, folgt

$$A \notin \mathrm{REC}$$

mit dem Unentscheidbarkeitskriterium. ❑

$\leq_{\mathrm{m}}$-vollständige Mengen in RE

Behauptung: Ist $X \leq_{\mathrm{m}} Y$ und X $\leq_{\mathrm{m}}$-vollständig in RE und $Y \in \mathrm{RE}$, so ist Y $\leq_{\mathrm{m}}$-vollständig in RE.

Beweis: ... ergibt sich unmittelbar aus der Transitivität der $\leq_{\mathrm{m}}$-Reduzierbarkeit. ❑

Das Postsche Korrespondenzproblem

Definition

Sei Σ ein Alphabet.

- Das *Postsche Korrespondenzproblem* (über Σ) ist definiert durch

$$\mathrm{PCP}_\Sigma = \left\{ ((x_1, y_1), \ldots, (x_k, y_k)) \;\middle|\; \begin{array}{l} k \in \mathbb{N} \text{ und } x_i, y_i \in \Sigma^+ \text{ für } 1 \leq i \leq k \\ \text{und es gibt } i_1, i_2, \ldots, i_n \in \{1, \ldots, k\}, \\ \text{so dass } x_{i_1} x_{i_2} \cdots x_{i_n} = y_{i_1} y_{i_2} \cdots y_{i_n} \end{array} \right\}.$$

- Eine solche Indexfolge $(i_1, i_2, \ldots, i_n)$ mit

$$x_{i_1} x_{i_2} \cdots x_{i_n} = y_{i_1} y_{i_2} \cdots y_{i_n}$$

heißt *Lösung der Probleminstanz* $((x_1, y_1), \ldots, (x_k, y_k))$.

Das modifizierte Postsche Korrespondenzproblem

- Das *„modifizierte" Postsche Korrespondenzproblem* (über Σ) ist definiert durch

$$\mathrm{MPCP}_\Sigma = \left\{ ((x_1, y_1), \ldots, (x_k, y_k)) \;\middle|\; \begin{array}{l} k \in \mathbb{N} \text{ und } x_i, y_i \in \Sigma^+ \text{ für } 1 \leq i \leq k \\ \text{und es gibt } i_2, \ldots, i_n \in \{1, \ldots, k\} \\ \text{so dass } x_1 x_{i_2} \cdots x_{i_n} = y_1 y_{i_2} \cdots y_{i_n} \end{array} \right\}.$$

  Das heißt, das MPCP_Σ ist die Einschränkung des PCP_Σ auf diejenigen Eingaben $((x_1, y_1), \ldots, (x_k, y_k))$, die eine mit dem Index 1 beginnende Lösung haben.

- Seien $\mathrm{PCP} = \bigcup_{\Sigma \text{ ist Alphabet}} \mathrm{PCP}_\Sigma$ und $\mathrm{MPCP} = \bigcup_{\Sigma \text{ ist Alphabet}} \mathrm{MPCP}_\Sigma$.

Das Postsche Korrespondenzproblem

Beispiel: Sei $\Sigma = \{0, 1\}$.

- Die Probleminstanz

$$\begin{pmatrix} x_1 & = & 1 & \quad & x_2 & = & 01 & \quad & x_3 & = & 010 \\ y_1 & = & 10 & \quad & y_2 & = & 1 & \quad & y_3 & = & 100 \end{pmatrix}$$

hat die Lösung $(1, 3, 3, 2)$, denn

$$\begin{array}{ccccccccccccccc} x_1 & x_3 & x_3 & x_2 & = & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ y_1 & y_3 & y_3 & y_2 & = & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}$$

und ist somit in PCP_Σ und in MPCP_Σ.

Das Postsche Korrespondenzproblem

- Andererseits hat die Probleminstanz

$$\begin{pmatrix} x_1 & = & 01 & \quad & x_2 & = & 010 & \quad & x_3 & = & 1 \\ y_1 & = & 1 & \quad & y_2 & = & 100 & \quad & y_3 & = & 10 \end{pmatrix}$$

zwar die Lösung $(3, 2, 2, 1)$ und ist damit in PCP_Σ, aber sie hat keine Lösung in MPCP_Σ, da $x_1 = 01$ und $y_1 = 1$.

Das Postsche Korrespondenzproblem

- Die Probleminstanz

$$\begin{pmatrix} x_1 & = & 001 & \quad & x_2 & = & 01 & \quad & x_3 & = & 01 & \quad & x_4 & = & 10 \\ y_1 & = & 0 & \quad & y_2 & = & 011 & \quad & y_3 & = & 101 & \quad & y_4 & = & 001 \end{pmatrix}$$

  ist lösbar, aber die kleinstmögliche lösende Indexfolge besteht aus $n = 66$ Elementen.

Das Postsche Korrespondenzproblem: $\mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}$

Lemma

$\mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}$.

Beweis: Wir zeigen

$$\mathrm{MPCP}_{\Sigma} \leq_{\mathrm{m}} \mathrm{PCP}_{\Sigma \cup \{\$, \#\}},$$

wobei $\$, \# \notin \Sigma$ neue Zeichen sind.

Für jedes Wort $w \in \Sigma^+$ mit $w = a_1 a_2 \cdots a_m$ definieren wir:

$$\begin{array}{lcccccccccc}
\overleftrightarrow{w} & = & \# & a_1 & \# & a_2 & \# & \cdots & \# & a_m & \# \\
\overrightarrow{w} & = & & a_1 & \# & a_2 & \# & \cdots & \# & a_m & \# \\
\overleftarrow{w} & = & \# & a_1 & \# & a_2 & \# & \cdots & \# & a_m &
\end{array}$$

Das Postsche Korrespondenzproblem: $\mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}$

Jeder Eingabe $z = ((x_1, y_1), \ldots, (x_k, y_k))$ des MPCP_Σ ordnen wir diese Eingabe des $\mathrm{PCP}_{\Sigma \cup \{\$,\#\}}$ mit $k + 2$ Komponenten zu:

$$f(z) = ((\overleftrightarrow{x}_1, \overleftarrow{y}_1), (\overrightarrow{x}_1, \overleftarrow{y}_1), \ldots, (\overrightarrow{x}_k, \overleftarrow{y}_k), (\$, \#\$)).$$

Offenbar ist die so definierte Funktion f in $\mathbb{R}$.

Zu zeigen ist, dass für alle z gilt:

$$z \in \mathrm{MPCP}_\Sigma \iff f(z) \in \mathrm{PCP}_{\Sigma \cup \{\$,\#\}}.$$

Das Postsche Korrespondenzproblem: $\mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}$

(⇒) Die Eingabe $z \in \mathrm{MPCP}_\Sigma$ habe die Lösung $(1, i_2, i_3, \ldots, i_n)$, d.h.,

$$x_1 x_{i_2} \cdots x_{i_n} = y_1 y_{i_2} \cdots y_{i_n}.$$

Bezeichnen wir das r-te Symbol von x_{i_j} mit $x_{i_j}^r$ bzw. das r-te Symbol von y_{i_j} mit $y_{i_j}^r$, dann gilt für $f(z)$:

$$\begin{aligned} f(z) &= \overbrace{\#x_1^1\#\cdots\#x_1^{r_1}\#}^{\overleftrightarrow{x}_1}\overbrace{x_{i_2}^1\#\cdots\#x_{i_2}^{r_2}\#}^{\overrightarrow{x}_{i_2}}\cdots\#\overbrace{x_{i_n}^1\#\cdots\#x_{i_n}^{r_n}\#}^{\overrightarrow{x}_{i_n}}\$ \\ &= \underbrace{\#y_1^1\#\cdots\#y_1^{s_1}}_{\overleftarrow{y}_1}\underbrace{\#y_{i_2}^1\#\cdots\#y_{i_2}^{s_2}}_{\overleftarrow{y}_{i_2}}\#\cdots\underbrace{\#y_{i_n}^1\#\cdots\#y_{i_n}^{s_n}}_{\overleftarrow{y}_{i_n}}\#\$ \end{aligned}$$

Somit ist $(1, i_2+1, i_3+1, \ldots, i_n+1, k+2)$ Lösung von $f(z)$ in $\mathrm{PCP}_{\Sigma\cup\{\$,\#\}}$.

Das Postsche Korrespondenzproblem: $\mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}$

($\Leftarrow$) Hat $f(z)$ die Lösung $(i_1, i_2, \ldots, i_n)$ in $\mathrm{PCP}_{\Sigma \cup \{\$,\#\}}$, so kann dies wegen der Bauart der Wortpaare nur gelten, wenn

1. $i_1 = 1$ und
2. $i_n = k + 2$ und
3. $i_j \in \{2, \ldots, k+1\}$ für $j \in \{2, \ldots, n-1\}$.

Folglich ist

$$(1, i_2 - 1, i_3 - 1, \ldots, i_{n-1} - 1)$$

eine Lösung von z in MPCP_{Σ}. ❑

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

Lemma

$H \leq_{\mathrm{m}} \mathrm{MPCP}$.

Beweis: Gegeben seien eine TM $M = (\Sigma, \Gamma, Z, \delta, z_0, \Box, F)$, geeignet codiert durch $\mathrm{code}(M) \in \{0,1\}^*$, und ein Eingabewort $w \in \Sigma^*$.

Wir codieren die Berechnung von $M(w)$ in eine Instanz des PCP.

Wir suchen also eine Reduktion $f \in \mathbb{R}$, so dass gilt:

$$M(w) \text{ hält in Endzustand} \tag{1}$$

$$\iff \quad f(\mathrm{code}(M)\#w) = ((x_1, y_1), \ldots, (x_k, y_k)) \text{ hat Lösung in } \mathrm{MPCP}_\Delta,$$

wobei $\Delta = \Gamma \cup Z \cup \{\#\}$ das Alphabet mit einem neuen Symbol $\#$ ist.

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

- Als erstes Paar wählen wir

$$(x_1, y_1) = (\#, \#z_0 w \#).$$

  Dabei ist $z_0 w$ die Startkonfiguration von $M(w)$.

- Die weiteren Paare

$$(x_i, y_i),$$

  $i > 1$, teilen wir ihrem Zweck entsprechend in Gruppen ein.

- **Kopierregeln:** (a, a) für alle $a \in \Gamma \cup \{\#\}$.

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

- **Überführungsregeln:**

$$
\begin{array}{rll}
(za, z'c) & \text{falls} & \delta(z,a) = (z',c,N) \\
(za, cz') & \text{falls} & \delta(z,a) = (z',c,R) \\
(bza, z'bc) & \text{falls} & \delta(z,a) = (z',c,L) \quad \text{für alle } b \in \Gamma \\
(\#za, \#z'\Box c) & \text{falls} & \delta(z,a) = (z',c,L) \\
(z\#, z'c\#) & \text{falls} & \delta(z,\Box) = (z',c,N) \quad \text{für } c \neq \Box \\
(z\#, z'\#) & \text{falls} & \delta(z,\Box) = (z',\Box,N) \\
(z\#, cz'\#) & \text{falls} & \delta(z,\Box) = (z',c,R) \\
(bz\#, z'bc\#) & \text{falls} & \delta(z,\Box) = (z',c,L) \quad \text{für alle } b \in \Gamma.
\end{array}
$$

- **Löschregeln:** (az, z) und (za, z) für alle $a \in \Gamma$ und $z \in F$.
- **Abschlussregeln:** $(z\#\#, \#)$ für alle $z \in F$.

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

Nun zeigen wir die Äquivalenz (1):

M hält bei Eingabe w im Endzustand

$\iff$ es gibt eine Folge $(k_0, k_1, \ldots, k_t)$ von Konfigurationen mit $k_0 = z_0 w$ und $k_t = uzv$ für $u, v \in \Gamma^*$, $z \in F$ und $k_{i-1} \vdash_M k_i$ für $1 \leq i \leq t$

$\iff$ $f(\mathrm{code}(M)\#w) = ((x_1, y_1), \ldots, (x_k, y_k))$ hat eine Lösung $(1, i_2, \ldots, i_n)$ für MPCP_Σ.

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

Das Lösungswort, das sich dabei ergibt, hat die Gestalt:

$$
\begin{array}{lllllllllllllllll}
x_1 & x_{i_2} & x_{i_3} & x_{i_4} & x_{i_5} & \dots & & & & & & & & & & x_{i_n} \\
\# & k_0 & \# & k_1 & \# & \dots & k_{t-1} & \# & k_t & \# & \# & k_t' & \# & k_t'' & \dots & z\#\# \\
\#k_0\# & k_1 & \# & k_2 & \# & \dots & k_t & \# & k_{t-1} & \# & k_t & \# & \# & k_t' & \dots & \# \\
y_1 & y_{i_2} & y_{i_3} & y_{i_4} & y_{i_5} & \dots & & & & & & & & & & y_{i_n}
\end{array}
$$

bzw.

$$
\begin{aligned}
x_1 x_{i_2} \cdots x_{i_n} &= \#k_0\#k_1\# \cdots \#k_t\#\#k_t^{'}\#k_t^{''}\# \cdots \#z\#\# \\
&= y_1 y_{i_2} \cdots y_{i_n},
\end{aligned}
$$

wobei die $k_t^{'}, k_t^{''}, \dots$ aus der Endkonfiguration $k_t = uzv$ durch sukzessives Löschen der Nachbarsymbole von $z \in F$ entstehen.

Die x_{i_j} „hinken“ dabei den y_{i_j} um genau eine Konfiguration nach.

Das Postsche Korrespondenzproblem: $H \leq_{\mathrm{m}} \mathrm{MPCP}$

Hat umgekehrt

$$f(\mathrm{code}(M)\#w) = ((x_1, y_1), \ldots, (x_k, y_k))$$

eine Lösung

$$(1, i_2, \ldots, i_n)$$

in MPCP_Σ, so kann aus dieser eine Rechnung von $M(w)$ abgelesen werden, die im Endzustand hält. ❑

MPCP und PCP sind $\leq_{\mathrm{m}}$-vollständig in RE

Theorem

Die Probleme MPCP *und* PCP *sind* $\leq_{\mathrm{m}}$*-vollständig in* RE *und somit unentscheidbar, d.h.,* $\mathrm{MPCP} \notin \mathrm{REC}$ *und* $\mathrm{PCP} \notin \mathrm{REC}$.

Beweis: Nach den obigen Lemmata und der $\leq_{\mathrm{m}}$-Vollständigkeit von H in RE gilt für alle $A \in \mathrm{RE}$:

$$A \leq_{\mathrm{m}} H \leq_{\mathrm{m}} \mathrm{MPCP} \leq_{\mathrm{m}} \mathrm{PCP}.$$

Dass MPCP und PCP in RE liegen, lässt sich einfach mit dem Projektionssatz zeigen. ❑

$\mathrm{PCP}_{\{0,1\}} \notin \mathrm{REC}$

Theorem

$\mathrm{PCP}_{\{0,1\}} \notin \mathrm{REC}$.

Beweis: Wir zeigen:

$$\mathrm{PCP}_{\Sigma} \leq_{\mathrm{m}} \mathrm{PCP}_{\{0,1\}}.$$

Das Alphabet von PCP sei $\Sigma = \{a_1, \ldots, a_n\}$.

Wir betrachten eine umkehrbare Codierung

$$f : \Sigma^* \to \{0,1\}^*$$

der Wörter über Σ mit der Eigenschaft, dass sich Wortketten eindeutig reproduzieren lassen.

$\mathrm{PCP}_{\{0,1\}} \notin \mathrm{REC}$

Ein Beispiel für eine solche Codierung von $\Sigma = \{a_1, \ldots, a_n\}$ ist die totale und berechenbare Funktion $f : \Sigma^* \to \{0,1\}^*$ mit

$$f(a_r) = 01^r \quad \text{und} \quad f(a_{i_1} \cdots a_{i_k}) = f(a_{i_1}) \cdots f(a_{i_k}).$$

An der Position der Nullen kann man eindeutig erkennen, dass dort die Codierung eines neuen Symbols beginnt und an der Anzahl der Einsen kann man erkennen, welches Symbol dort codiert ist.

Es gilt:

$$((x_1, y_1), \ldots, (x_k, y_k)) \text{ hat eine Lösung}$$
$$\Leftrightarrow$$
$$((f(x_1), f(y_1)), \ldots, (f(x_k), f(y_k))) \text{ hat eine Lösung.}$$

❑

$\mathrm{PCP}_{\{1\}} \in \mathrm{REC}$

Theorem

Für jedes Σ mit $|\Sigma| = 1$ ist PCP_Σ entscheidbar.

Beweis: Siehe Vorlesungsskript. ❑

Bemerkung: Definiert man k-PCP als die Einschränkung von PCP, in der für ein festes $k \in \mathbb{N}$ höchstens k-Tupel $((x_1, y_1), \ldots, (x_k, y_k))$ auftreten, so ist bekannt, dass

- k-PCP ist entscheidbar für $k \leq 2$, aber
- k-PCP ist nicht entscheidbar für $k \geq 9$.

Offen ist die Frage nach der Entscheidbarkeit von k-PCP für $3 \leq k \leq 8$.

(Un)Entscheidbarkeit in der Chomsky-Hierarchie

Definition

Für $i \in \{0, 1, 2, 3\}$ definieren wir das *Wortproblem*, das *Leerheitsproblem*, das *Schnittproblem*, das *Äquivalenzproblem* und das *Mehrdeutigkeitsproblem* für Typ-i-Grammatiken:

$$\begin{aligned}
\text{Wort}_i &= \{(G, w) \mid G \text{ ist Typ-}i\text{-Grammatik und } w \in L(G)\},\\
\text{Leer}_i &= \{G \mid G \text{ ist Typ-}i\text{-Grammatik und } L(G) \neq \emptyset\},\\
\text{Schnitt}_i &= \{(G_1, G_2) \mid G_1,\ G_2 \text{ sind Typ-}i\text{-Grammatiken und } L(G_1) \cap L(G_2) \neq \emptyset\},\\
\text{Äquiv}_i &= \{(G_1, G_2) \mid G_1,\ G_2 \text{ sind Typ-}i\text{-Grammatiken und } L(G_1) = L(G_2)\},\\
\text{Ambig}_i &= \{G \mid G \text{ ist eine mehrdeutige Typ-}i\text{-Grammatik}\}.
\end{aligned}$$

Statt der Grammatiken in den Problemen können auch jeweils äquivalente Automatenmodelle eingesetzt werden.

(Un)Entscheidbarkeit in der Chomsky-Hierarchie

Theorem

Die Tabelle zeigt die Entscheidbarkeit bzw. Nicht-Entscheidbarkeit für die oben definierten Probleme in der Chomsky-Hierarchie.

i	Wort_i	Leer_i	Schnitt_i	Äquiv_i	Ambig_i
0	$\notin$ REC	$\notin$ REC	$\notin$ REC	$\notin$ REC	$\notin$ REC
1	$\in$ REC	$\notin$ REC	$\notin$ REC	$\notin$ REC	$\notin$ REC
2	$\in$ REC	$\in$ REC	$\notin$ REC	$\notin$ REC	$\notin$ REC
3	$\in$ REC	$\in$ REC	$\in$ REC	$\in$ REC	$\in$ REC

Tabelle: *Entscheidbarkeit von Problemen in der Chomsky-Hierarchie*

Beweisideen: Wortproblem

- Man kann zeigen, dass

$$H \leq_{\mathrm{m}} \mathrm{Wort}_0$$

  gilt und damit folgt, dass Wort_0 nicht entscheidbar ist.

- Wort_1 ist entscheidbar wegen $\mathrm{CS} \subseteq \mathrm{REC}$.

- Wort_2 ist entscheidbar, nach dem sog. CYK-Algorithmus.

- Wort_3 ist entscheidbar; einfach über DFAs.

Beweisideen: Leerheitsproblem

- Man kann zeigen, dass

$$H_0 \leq_{\mathrm{m}} \mathrm{Leer}_1$$

  gilt. Damit ist Leer_1 nicht entscheidbar.

- Da jede Typ-1-Grammatik auch eine Typ-0-Grammatik ist, folgt dass auch Leer_0 nicht entscheidbar ist.

- Zur Entscheidbarkeit von Leer_2: Es sei G eine kontextfreie Grammatik mit k Nichtterminalen. Sei n die Zahl aus dem Pumping-Lemma für $L(G)$, nämlich $n = 2^{k+1}$.

Beweisideen: Leerheitsproblem

Wir zeigen:

$$L(G) \neq \emptyset \quad \Longleftrightarrow \quad \text{es gibt ein Wort } z \in L(G) \text{ mit } |z| < n. \qquad (2)$$

($\Leftarrow$) Klar.

($\Rightarrow$) Sei $L(G) \neq \emptyset$ und sei z ein Wort minimaler Länge in $L(G)$. Wäre $|z| \geq n$, dann folgte aus dem Pumping-Lemma: $z = uvwxy$ mit $|vx| \geq 1$, $|vwx| \leq n$ und $(\forall i \geq 0)\,[uv^iwx^iy \in L]$. Insbesondere wäre $uwy \in L(G)$ für $i = 0$, und es wäre $|uwy| < |uvwxy| = |z|$ ein Widerspruch zur Minimalität von z. Also ist $|z| < n$. Die Äquivalenz ist bewiesen.

Da die rechte Seite der Äquivalenz entscheidbar ist, folgt Leer_2 ist entscheidbar.

Beweisideen: Leerheitsproblem

- Damit muss auch Leer_3 entscheidbar sein, da jede Typ-3-Grammatik auch eine Typ-2-Grammatik ist.

Beweisideen: Schnittproblem

- Man kann zeigen, dass

$$\text{PCP} \leq_{\text{m}} \text{Schnitt}_2$$

  gilt. Damit ist Schnitt_2 nicht entscheidbar.

- Damit sind auch Schnitt_1 und Schnitt_0 nicht entscheidbar.

- Schnitt_3 ist entscheidbar, da man einen so genannten Kreuzproduktautomaten angeben kann, der $L(G_1) \cap L(G_2)$ akzeptiert.

Beweisideen: Äquivalenzproblem

- Man kann zeigen, dass

$$\text{Schnitt}_2 \leq_{\text{m}} \text{Äquiv}_2$$

gilt. Damit ist Äquiv_2 nicht entscheidbar.

- Damit sind auch Äquiv_1 und Äquiv_0 nicht entscheidbar.

- Äquiv_3 ist entscheidbar, z.B. durch Testen der Minimalautomaten von $L(G_1)$ und $L(G_2)$ auf Isomorphie.

  Oder durch das entscheidbare Leerheitsproblem Leer_3:

$$L(G_1) = L(G_2) \Leftrightarrow (L(G_1) \cap \overline{L(G_2)}) \cup (\overline{L(G_1)} \cap L(G_2)) = \emptyset.$$

Beweisideen: Mehrdeutigkeit für kfGs

- Man kann $\mathrm{PCP} \leq_m \mathrm{Ambig}_2$ wie folgt zeigen:
- PCP-Instanz $((x_1, y_1), \ldots, (x_k, y_k))$ umwandeln in die folgende kontextfreie Grammatik:
  - $S \to A \mid B$
  - $A \to 1Ax_1 \mid 2Ax_2 \mid \ldots \mid kAx_k \mid \$$
  - $B \to 1By_1 \mid 2By_2 \mid \ldots \mid kBy_k \mid \$$
- Man kann zeigen, dass diese Grammatik genau dann mehrdeutig ist, wenn die gegebene PCP-Instanz lösbar ist.