ad Proof Thm 2.1.4 in Chap 4

Compare Bsp 4.4.1(a) in Winkler, Formale Grundlagen 2

	Liste A	Liste B
i	w_i	x_i
1	1	111
2	10111	10
3	10	0

this instance of PCP is solvable:

$w_2w_1w_1w_3 = 101111110 = x_2x_1x_1x_3.$

Corresponding string-rewriting system $R = R_1 \cup R_2$:

R_1: 1: $(A, 1A01)$, 2: $(1A01, 1CC01)$
3: $(A, 10111A001)$, 4: $(10111A001, 10111CC001)$
5: $(A, 10A0001)$, 6: $(10A0001, 10CC0001)$

R_2: 1: $(B, 111B01)$, 2: $(111B01, 111CC01)$
3: $(B, 10B001)$, 4: $(10B001, 10CC001)$
5: $(B, 0B0001)$, 6: $(0B0001, 0CC0001)$

Then we have

$A \to_{R_{13}} 10111A001 \to_{R_{11}} 101111A01001$

$\to_{R_{11}} 1011111A0101001 \to_{R_{15}} 101111110A00010101001$

$\to_{R_{16}} 10111111 0 CC 0001 0101001$

$\leftarrow_{R_{22}} 10111111 0 B 0001 0101001 \leftarrow_{R_{25}} 1011111 B 01 01001$

$\leftarrow_{R_{21}} 10111 B 01001 \leftarrow_{R_{21}} 10 B 001 \leftarrow_{R_{23}} B$

4.4. Das Korrespondenzproblem von Post

Ein anderes unentscheidbares Problem ist das Korrespondenzproblem von Post. Die Unentscheidbarkeit des Korrespondenzproblems von Post hat zur Folge, daß auch viele andere interessante Problem als unentscheidbar erwiesen werden können, so z.B. die Mehrdeutigkeit kontextfreier Grammatiken.

Definition 4.4.1: (a) Eine *Instanz des Korrespondenzproblems von Post (Post's correspondence problem, PCP)* besteht aus zwei Listen von Wörtern

$$A = w_1, \ldots, w_k \qquad \text{und} \qquad B = x_1, \ldots, x_k$$

über einem Alphabet Σ. Diese Instanz des PCP *hat eine Lösung* genau dann, wenn es eine Folge von ganzen Zahlen $i_1, \ldots, i_m \in \{1, \ldots, k\}$, mit $m \geq 1$, gibt, sodaß

$$w_{i_1} \cdots w_{i_m} = x_{i_1} \cdots x_{i_m}.$$

Die Folge $i_1, \ldots, i_m$ ist eine *Lösung* dieser Instanz des PCP.

(b) Ein *Instanz des modifizierten Korrespondenzproblems von Post (MPCP)* sieht aus wie eine Instanz des PCP. Für eine *Lösung* des MPCP muß aber zusätzlich gelten $i_1 = 1$, also der Lösungsstring muß jeweils mit dem ersten Element der Listen A, B beginnen. ■

Beispiel 4.4.1: Sei das zugrunde liegende Alphabet $\Sigma = \{0, 1\}$.

(a) Wir betrachten die Instanz des PCP, welche durch folgende Listen gegeben ist:

	Liste A	Liste B
i	w_i	x_i
1	1	111
2	10111	10
3	10	0

Diese Instanz des PCP hat eine Lösung, nämlich

$$m = 4, i_1 = 2, i_2 = 1, i_3 = 1, i_4 = 3,$$

also

$$w_2 w_1 w_1 w_3 = 101111110 = x_2 x_1 x_1 x_3.$$

(b) Wir betrachten die Instanz des PCP, welche durch folgende Listen gegeben ist:

	Liste A	Liste B
i	w_i	x_i
1	10	101
2	011	11
3	101	011

Angenommen $i_1, \ldots, i_m$ ist eine Lösung dieser Instanz des PCP. Um die Übereinstimmung am Stringanfang zu gewährleisten muß gelten $i_1 = 1$. Also wir haben bisher als Teillösung

$$w_1 = 10$$
$$x_1 = 101$$

Wir müssen nun aus A ein Wort wählen, das mit 1 beginnt, also $i_2 \in \{1, 3\}$. $i_2 = 1$ funktioniert aber nicht, da wir sonst in der Teillösung eine Diskrepanz an der vierten Stelle hätten. Somit bleibt nur $i_2 = 3$, also die Teillösung

$$w_1 w_3 = 10101$$
$$x_1 x_3 = 101011$$

Nun sind wir aber wieder in der gleichen Situation wie vorhin, nämlich daß die Teillösung aus B um eine 1 länger ist als die Teillösung aus A. Mit derselben Argumentation wie oben ergibt sich also $i_3 = i_4 = \ldots = 3$. Eine andere Wahlmöglichkeit gibt es nicht. Die beiden Strings können also nie dieselbe Länge haben. Somit ist diese Instanz des PCP unlösbar. ■

Satz 4.4.1: *Wäre* PCP *entscheidbar, dann wäre auch* MPCP *entscheidbar.*

Beweis: Sei

$$A = w_1, \ldots, w_k, \qquad B = x_1, \ldots, x_k$$

eine Instanz von MPCP. Wir transformieren diese Instanz von MPCP in eine Instanz von PCP, welche genau dann lösbar ist, wenn die gegebene Instanz von MPCP lösbar ist. Damit ist der Satz dann bewiesen.

Sei Σ das kleinste Alphabet, welches alle Symbole in A und B enthält und zusätzlich die beiden Symbole $\not{o}$ und \$. Sei y_i das Wort, das aus w_i entsteht, indem man nach jedem Symbol in w_i das Symbol $\not{o}$ einfügt, und sei z_i das Wort, das aus x_i entsteht, indem man vor jedem Symbol in x_i das Symbol $\not{o}$ einfügt. Weiters betrachten wir neue Wörter

$$y_0 = \not{o} y_1, \qquad z_0 = z_1,$$

$$y_{k+1} = \$, \qquad z_{k+1} = \emptyset\$.$$

Seien nun

$$C = y_0, y_1, \ldots, y_{k+1}, \qquad D = z_0, z_1, \ldots, z_{k+1}.$$

Für die Listen aus Beispiel 4.4.1(a) ergibt sich etwa dadurch folgende Transformation:

MPCP:

	Liste A	Liste B
i	w_i	x_i
1	1	111
2	10111	10
3	10	0

PCP:

	Liste C	Liste D
i	y_i	z_i
0	ø1ø	ø1ø1ø1
1	1ø	ø1ø1ø1
2	1ø0ø1ø1ø1ø	ø1ø0
3	1ø0ø	ø0
4	\$	ø\$

Die Listen C und D spezifizieren eine Instanz des PCP, welche genau dann lösbar ist, wenn die gegebene Instanz des MPCP lösbar ist.

Denn ist etwa $1, i_1, \ldots, i_r$ eine Lösung des MPCP, dann ist $0, i_1, \ldots, i_r, k+1$ eine Lösung des PCP.

Andererseits, ist $i_1, \ldots, i_r$ eine Lösung des PCP, dann muß gelten $i_1 = 0$ und $i_r = k+1$. Sei j der kleinste Index sodaß $i_j = k+1$. Dann ist $i_1, \ldots, i_j$ auch eine Lösung, da das Symbol \$ nur als letztes Symbol von y_{k+1} und z_{k+1} vorkommt, und für kein l mit $1 \leq l < j$ gilt $i_l = k+1$. Somit ist $1, i_2, \ldots, i_{j-1}$ offenbar eine Lösung des MPCP.

Gibt es also einen Algorithmus um PCP zu entscheiden, so erhalten wir daraus auch einen Algorithmus um MPCP zu entscheiden. ■

Mit dieser Hilfsüberlegung sind wir nun in der Lage, die Unentscheidbarkeit von PCP zu beweisen.

Satz 4.4.2: PCP *ist unentscheidbar.*

Beweis: Wir zeigen, daß das Akzeptierungsproblem für Turing–Maschinen auf MPCP reduzierbar ist, d.h. wäre MPCP entscheidbar, so wäre auch das Akzeptierungsproblem entscheidbar. Das ist aber laut Satz 4.1.3 nicht der Fall. Somit ist MPCP nicht entscheidbar, und wegen Satz 4.4.1 auch PCP nicht entscheidbar.

Für jede TM M und jedes Wort w konstruieren wir ein MPCP, welches lösbar ist genau dann, wenn es eine Lösung der Form

$$\#q_0w\#u_1q_1v_1\#\cdots\#u_kq_kv_k\#\cdots$$

hat, wobei Strings zwischen aufeinanderfolgenden # aufeinanderfolgende Konfigurationen von M sind bei Eingabe w, und q_k ein akzeptierender Zustand von M ist.

Wir geben nun die Listen A und B des zugehörigen MPCP an. Wir nehmen an, daß die TM M stoppt, sobald ein akzeptierender Zustand erreicht wird. Außer für das erste Paar vergeben wir keine Nummern, da diese für die Existenz einer Lösung irrelevant sind.

Gruppe 0: erstes Paar

Liste A	Liste B
$\#$	$\#q_0w\#$

Gruppe I: (Kopieren) für $\alpha \in \Gamma$:

Liste A	Liste B
α	α
$\#$	$\#$

Gruppe II: (Anwenden der Überführungsfunktion) für $q \in Q \setminus F$, $p \in Q$, $\alpha, \beta, \gamma \in \Gamma$:

Liste A	Liste B	
$q\alpha$	βp	falls $\delta(q, \alpha) = (p, \beta, R)$
$\gamma q\alpha$	$p\gamma\beta$	falls $\delta(q, \alpha) = (p, \beta, L)$
$q\#$	$\beta p\#$	falls $\delta(q, \sqcup) = (p, \beta, R)$
$\gamma q\#$	$p\gamma\beta\#$	falls $\delta(q, \sqcup) = (p, \beta, L)$

Gruppe III: (Endbehandlung) für $q \in F$, und $\alpha, \beta \in \Gamma$:

Liste A	Liste B
$\alpha q\beta$	q
αq	q
$q\beta$	q

Gruppe IV: (Liste A abschließen) für $q \in F$:

Liste A	Liste B
$q\#\#$	$\#$

Wir nennen (x, y) eine *Teillösung* von MPCP bzgl. der Listen A und B, wenn x ein Anfangsstück (Prefix) von y ist, und x bzw. y durch Verkettung korrespondierender Strings aus den Listen A bzw. B hervorgehen. Ist $xz = y$, so nennen wir z den *Rest* von (x, y).

Angenommen ausgehend von der Konfiguration q_0w gibt es eine gültige Folge von k weiteren Konfigurationen, also eine Berechnung dieser Länge (die aber nicht notwendigerweise hier zu Ende sein muß). Dann behaupten wir, daß es eine Teillösung der Form

$$(x, y) = (\#q_0w\#u_1q_1v_1\#\cdots\#u_{k-1}q_{k-1}v_{k-1}\#,$$
$$\#q_0w\#u_1q_1v_1\#\cdots\#u_{k-1}q_{k-1}v_{k-1}\#u_kq_kv_k\#)$$

gibt. Zudem ist das die einzige Teillösung, deren längerer String so lang ist wie $|y|$.

Wir beweisen diese Behauptung durch Induktion über k. Für $k = 0$ ist die Behauptung trivial, da das Paar $(\#, \#q_0w\#)$ als erstes gewählt werden muß.

Angenommen die Behauptung gilt für ein k, und daß q_k nicht in F ist. Der Rest des Paares (x, y) ist $z = u_kq_kv_k\#$. Die nächsten Paare müssen nun so gewählt werden, daß ihre Anteile aus A den String z ergeben. Unabhängig davon welche Symbole rechts und links von q_k stehen, gibt es höchstens ein Paar in Gruppe II, welches es erlaubt, die Teillösung über q_k hinaus fortzusetzen. Dieses Paar entspricht in natürlicher Weise einem Berechnungsschritt von M ausgehend von der Konfiguration $u_kq_kv_k$. Die anderen Symbole von z führen zwangsläufig zu Paaren aus der Gruppe I. Keine andere Auswahl von Paaren erlaubt es, z darzustellen als Verkettung von Elementen der Liste A. Somit erhalten wir eine neue Teillösung $(y, yu_{k+1}q_{k+1}v_{v+1}\#)$. Offensichtlich ist $u_{k+1}q_{k+1}v_{k+1}$ die einzige Konfiguration, welche M in einem Schritt von $u_kq_kv_k$ erreichen kann. Es gibt auch keine andere Teillösung, deren zweiter String so lang ist wie $|yu_{k+1}q_{k+1}v_{k+1}|$. Somit ist die Behauptung auch für $k + 1$ nachgewiesen.

Ist nun $q_k \in F$, so findet man leicht Paare aus den Gruppen I und III, welche es erlauben die Teillösung (x, y) zu einer Lösung des MPCP mit Listen A und B zu vervollständigen, nachdem als letztes Paar dasjenige aus Gruppe IV gewählt wurde.

Somit ist klar, daß MPCP mit Listen A und B eine Lösung besitzt, wenn M ausgehend von der Konfiguration q_0w einen akzeptierenden Zustand erreicht. Erreicht M keinen akzeptierenden Zustand, so können die Paare aus den Gruppen III und IV nicht angewendet werden. Somit wird in jeder Teillösung der zweite String länger sein als der erste, und es ist daher keine vollständige Lösung möglich.

Abschließend stellen wir also fest, daß die Instanz des MPCP genau dann eine Lösung hat, wenn M bei Eingabe w in einem akzeptierenden Zustand hält. Die obige Konstruktion kann für jede TM M und Eingabewort w ausgeführt werden. Gäbe es also einen Algorithmus zur Entscheidung von MPCP, dann gäbe es auch einen Algorithmus zur Entscheidung des Akzeptierungsproblems für Turing–Maschinen, im Widerspruch zu Satz 4.1.3. ∎

Übung: Sei M die TM

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, \sqcup\}, q_0, \{q_2\}, \delta),$$

wobei δ wie folgt definiert ist:

	0	1	$\sqcup$
q_0	$(q_1, 1, R)$	$(q_1, 0, L)$	$(q_1, 1, L)$
q_1	$(q_2, 0, L)$	$(q_0, 0, R)$	$(q_1, 0, R)$
q_2	—	—	—

Wie sieht die zugehörige Instanz des MPCP aus? M akzeptiert das Wort $w = 01$ in 4 Berechnungsschritten. Wie sieht die zugehörige Lösung der Instanz des MPCP aus?