Friedhelm Meyer auf der Heide
Dominic Dumrauf, Michael Kortenjan, Ulf Lorenz,
Gunnar Schomaker, Tim Süss, Mario Vodisek, Alexander Willms

Paderborn, den 11. Januar

Übung zur Vorlesung

Einführung in Berechenbarkeit, Komplexität und Formale Sprachen

WS 2004/05

Musterlösung zu Blatt 11

Aufgabe: 41

1. Die Sprache L_1 ist entscheidbar, denn die folgende DTM M_1 entscheidet L_1.

   M_1 : Bei Eingabe $x \in \Sigma^*$
   1. Falls x nicht von der Form $(\langle M\rangle, w, d)$ ist, lehne ab.
   2. Simuliere M mit Eingabe w für d Schritte.(Benutze universelle DTM und Zähler).
   3. Wird in 2. festgestellt, dass M die Eingabe w innerhalb von d Schritten akzeptiert, akzeptiere $(\langle M\rangle, w, d)$.
   4. Wird in 2. festgestellt, dass M die Eingabe w nicht innerhalb von d Schritten akzeptiert, lehne $(\langle M\rangle, w, d)$ ab.

2. Die Sprache L_2 ist rekursiv aufzählbar. Um dieses zu beweisen, konstruieren wir eine DTM M_2, die L_2 akzeptiert.

   M_2 : Bei Eingabe $x \in \Sigma^*$
   1. Falls x nicht von der Form $(\langle M\rangle, w, d)$ ist, lehne ab.
   2. Simuliere zunächst M mit Eingabe w für d Schritte. (Universelle DTM und Zähler).
   3. Wird in 2. festgestellt, dass M die Eingabe w innerhalb von d Schritten akzeptiert oder ablehnt, lehne $(\langle M\rangle, w, d)$ ab.
   4. Simuliere M weiter. Wird hierbei festgestellt, dass M die Eingabe w akzeptiert, akzeptiere $(\langle M\rangle, w, d)$.

   Schritt 4 wird nur von Eingaben $(\langle M\rangle, w, d)$ erreicht, bei denen M bei Eingabe w mehr als d Schritte benötigt. Unter den Tripeln, die in 3. nicht abgelehnt werden, werden dann in 4. nur solche akzeptiert, bei denen M die Eingabe w nach mindestens d Schritten akzeptiert. Damit akzeptiert M_2 die Sprache L_2.

   Die DTM M_2 hält nicht bei jeder Eingabe. Gerät die DTM M bei Eingabe w in eine Endlosschleife, dann wird M_2 bei Eingabe $(\langle M\rangle, w, d)$ nicht halten. Damit entscheidet M_2 die Sprache L_2 nicht. (L_2 ist nicht entscheidbar; der Beweis ist jedoch nicht Bestandteil dieser Aufgabe.)

Aufgabe 42:

Das Totalitätsproblem $T = \{\langle M\rangle \mid L(M) = \Sigma^*\}$ ist nach Vorlesung nicht entscheidbar. Wir reduzieren $T \leq L$ und definieren folgende Reduktionsfunktion f:

$$f(w) = \begin{cases} \langle M_1\rangle\langle M\rangle & \text{,falls } w = \langle M\rangle \text{ eine Gödelisierung ist} \\ \langle M_2\rangle\langle M_2\rangle & \text{,sonst} \end{cases}$$

Dabei sind M_1 und M_2 feste DTMs, $L(M_1) = \Sigma^* \setminus \{0\}$, $L(M_2) = \Sigma^*$.
Es ist einfach, solche DTM anzugeben.

Zu zeigen: 1. f ist berechenbar (ist klar),
2. $w \in T \Leftrightarrow f(w) \in L$

Beweis zu 2:

$$\begin{aligned} w \in T &\Rightarrow w = \langle M\rangle, \text{ mit } L(M) = \Sigma^* \\ &\Rightarrow L(M_1) \subset L(M),\ L(M_1) = \Sigma^* \setminus \{0\} \subset \Sigma^* = L(M) \\ &\Rightarrow f(w) = \langle M_1\rangle\langle M\rangle \in L \end{aligned}$$

$w \notin T \Rightarrow$ w ist keine Gödelisierung oder $w = \langle M\rangle$ und $L(M) \neq \Sigma^*$
$\Rightarrow f(w) = \langle M_2\rangle\langle M_2\rangle \notin L$
oder $f(w) = \langle M_1\rangle\langle M\rangle$. Da $L(M) \neq \Sigma^*$ ist, kann $L(M_1) = \Sigma^* \setminus \{0\}$ nicht echte Teilmenge von $L(M)$ sein. Also ist $f(w) = \langle M_1\rangle\langle M\rangle \notin L$.

Aufgabe 44:

$\Rightarrow$
Sei L NP-vollständig. Da NP=Co-NP ist, sind auch L und $\bar{L}$ in NP.
$\Leftarrow$
Sei L NP-vollständig, $\bar{L} \in$ NP.
Sei $L' \in$ NP. Dann gilt: $L' \leq_p L$.
Beh: Dann gilt aber auch: $\bar{L}' \leq_p \bar{L}$.

Bew: Sei $L' \leq_p L$ mittels f.
Dann gilt $\forall x \in \Sigma^*$: $x \in L' \Leftrightarrow f(x) \in L$.
Das ist äquivalent zu: $\forall x \in \Sigma^*$: $x \in \bar{L}' \Leftrightarrow f(x) \in \bar{L}$. $\square$

Somit entscheidet folgende NTM $\tilde{M}$ $\bar{L}'$ in polynomieller Zeit:
Bei Eingabe $x \in \Sigma^*$:

- Berechne f(x)
- Nutze polynomiell zeitbeschränkte NTM für $\bar{L}$, um „$f(x) \in \bar{L}$?“ zu testen.

Nach oben gezeigtem ist $\tilde{M}$ eine polynomiell zeitbeschränkte NTM für $\bar{L}'$, also: $\bar{L}' \in$ NP, also $L' \in$ Co-NP.

Somit ist NP $\subseteq$ Co-NP. (i)

NP $\supseteq$ Co-NP gilt, da Co-NP $\overset{(i)}{\subseteq}$ Co-(Co-NP) $=$ NP.

Aufgabe 45:

Wir müssen eine Reduktionsfunktion f angeben, die für jede Eingabe $(\langle G\rangle, k)$ ein $f(\langle G\rangle, k) = A, b$ erzeugt mit:
G enthält eine unabhängige Menge der Größe $k \Leftrightarrow Ax \leq b$ hat eine 0-1-Lösung x.

Wir konstruieren f wie folgt:
Sei $(\langle G\rangle, k)$ eine Instanz für IS. G habe Knotenmenge $\{1, \ldots, n\}$. Dann hat das zugehörige lineare Programm Variablen $x_1, \ldots, x_n$, wobei „$x_i = 1$“ bedeuten soll: „i ist in der unabhängigen Menge“. Das Ungleichungssystem sieht wie folgt aus:

$$\begin{array}{rcl} \sum_{i=1}^{n} x_i & \leq & k \\ \sum_{i=1}^{n} -x_i & \leq & -k \\ x_i + x_j & \leq & 1 \ \forall \{i,j\} \in E(G) \end{array}$$

Der Aufwand zur Berechnung dieses Ungleichungssystems aus $(\langle G\rangle, k)$ ist offensichtlich polynomiell. Die Äquivalenz für $x \in$ IS $\Leftrightarrow f(x) \in$ BP folgt offensichtlich aus der Konstruktion.

Aufgabe 46

Sei k die Anzahl der Kartons, die Algorithmus Eiliges Packen benötigt. Mit $g(K_i)$ bezeichnen wir die Gesamtgrösse der Gegenstände, die Algorithmus Eiliges Packen in den i-ten Karton K_i liegt. Nun gilt

$$g(K_i) + g(K_j) > 1 \quad \text{für alle} \quad 1 \leq i < j \leq k. \tag{1}$$

Wäre (1) nicht erfüllt, so hätte Eiliges Packen, die Gegenstände in Karton K_j in einen früher geöffneten Karton, nämlich spätestens in Karton K_i gelegt.
Aus (1) folgt nun

$$\sum_{i=1}^{n} a_i > \lfloor \frac{k}{2} \rfloor. \tag{2}$$

Sei nun opt die minimale Anzahl von Kartons, die zum Verstauen aller i Gegenstände benötigt werden. Da in (1) echte Ungleichheit gilt, folgt

$$\text{opt} \geq \lfloor \frac{k}{2} \rfloor + 1, (\text{da opt } \geq \sum_{i=1}^{n} a_i \text{ und opt} \in \mathbb{N} \text{ ist.})$$

Da $\lfloor \frac{k}{2} \rfloor + 1 \geq \frac{k}{2}$ erzielt Eiliges Packen damit bei jeder Instanz einen Approximationsfaktor von höchstens

$$\frac{k}{\frac{k}{2}} = 2.$$