Theoretische Informatik Übung 10 — W11 — Rita Nagy

Aufg 1) (Worked Example 3)

Sei $L_{all} := \{Kod(M) \mid M \text{ akzeptiert jede Eingabe}\}$

Zz: $L_H^c \leq_{EE} L_{all}$ (& Korrektheit der Reduktion)

Lösung: in der Übungsstunde besprochen.

6.3 Komplexitätsklassen und die Klasse P

Def: Für alle Funktionen $f, g: \mathbb{N} \to \mathbb{R}^+$ definieren wir

- $TIME(f) := \{L(B) \mid B \text{ ist eine MTM mit } Time_A(n) \in O(f(n))\}$
- $SPACE(g) := \{L(A) \mid A \text{ ist eine MTM mit } Space_A(n) \in O(g(n))\}$
- $DLOG := SPACE(\log_2(n))$
- $P := \bigcup_{c \in \mathbb{N}} TIME(n^c)$ (≙ alle Probleme, die in Polynomzeit lösbar sind); "praktisch entscheidbare Probleme"
- $PSPACE := \bigcup_{c \in \mathbb{N}} SPACE(n^c)$ (≙ ——"—— Polynomspeicher —"—)
- $EXPTIME := \bigcup_{d \in \mathbb{N}} TIME(2^{n^d})$

Lem 6.3: $\forall t: \mathbb{N} \to \mathbb{R}^+$ Fkt. gilt:

$$TIME(t(n)) \subseteq SPACE(t(n))$$

Beweis: $\forall M$ MTM: $Space_M(n) \leq Time_M(n)$, da M nicht mehr Felder pro AB beschriften kann, als sie Zeit beansprucht.

Kor 6.1: $P \subseteq PSPACE$

Satz 6.2: $\forall s: \mathbb{N} \to \mathbb{R}^+$ mit $s(n) \geq \log_2(n)$ gilt:

$$SPACE(s(n)) \subseteq \bigcup_{c \in \mathbb{N}} TIME(c^{s(n)})$$ "maximal polynomieller blow-up"

Fundamentale Hierarchie deterministischer Komplexitätsklassen: $DLOG \subseteq P \subseteq PSPACE \subseteq EXPTIME$

6.4 Nichtdeterministische Komplexitätsmasse

Def: Für alle Funktionen $f, g : \mathbb{N} \to \mathbb{R}^+$ definieren wir

- $NTIME(f) := \{L(B) \mid B$ ist eine NMTM mit $Time_A(n) \in O(f(n))\}$
- $NSPACE(g) := \{L(A) \mid A$ ist eine NMTM mit $Space_A(n) \in O(g(n))\}$
- $NLOG := NSPACE(\log_2(n))$
- $NP := \bigcup_{c \in \mathbb{N}} NTIME(n^c)$ (= alle Probleme, die in Polynomzeit lösbar sind);
- $NPSPACE := \bigcup_{c \in \mathbb{N}} NSPACE(n^c)$ (= —"— Polynomspeicher —"—)

Bmk: Aus Zeitgründen skippen wir einige (wichtige) Resultate, die ihr im Buch nachlesen könnt.

Satz 6.7: (Satz von Savitch)
Sei $s : \mathbb{N} \to \mathbb{R}^+$ platzkonstr. mit $s(n) \geq n$. Dann gilt:
$NSPACE(s(n)) \subseteq SPACE(s(n)^2)$

Kor 6.5: PSPACE = NPSPACE

Fundamentale Komplexitätsklassenhierarchie der sequentiellen Berechnungen:
$DLOG \subseteq NLOG \subseteq P \subseteq NP \subseteq PSPACE \subseteq EXPTIME$

Bmk: Echtheit der Inklusionen ist überall unbekannt,
wir wissen aber $DLOG \subsetneq PSPACE$ und $P \subsetneq EXPTIME$

6.6 NP-Vollständigkeit

Def: Seien $L_1 \subseteq \Sigma_1^*$, $L_2 \subseteq \Sigma_2^*$ zwei Sprachen.

<u>L_1 ist polynomiell auf L_2 reduzierbar</u>, $L_1 \leq_p L_2$,

falls eine polynomielle TM (Algo) A existiert, die für jedes $x \in \Sigma_1^*$ ein $A(x) \in \Sigma_2^*$ berechnet, s.d.

$x \in L_1 \iff A(x) \in L_2$

A wird eine <u>polynomielle Reduktion</u> von L_1 auf L_2 genannt.

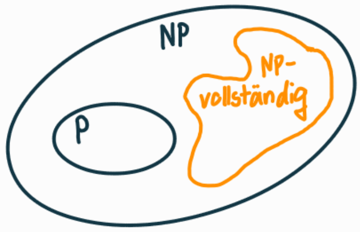
Bmk: Die Reduktion $\leq_p$ entspricht einer EE-Reduktion ($\leq_{EE}$) mit zusätzlicher Bedingung bzgl. der Effizienz.

Def: • Eine Sprache L ist <u>NP-schwer</u>, falls $\forall L' \in NP: \; L' \leq_p L$.

• Eine Sprache L ist <u>NP-vollständig</u>, falls

(i) $L \in NP$

(ii) L ist NP-schwer



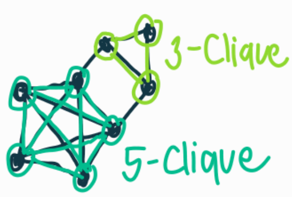
Bmk: Wir treffen die Annahme $P \subsetneq NP$. (sonst wäre die Menge der NP-vollständigen Sprachen leer.)

Lem 6.7: Falls $L \in P$ und L ist NP-schwer, dann gilt $P = NP$.

Lem 6.8: Seien L_1, L_2 Sprachen.
Falls $L_1 \leq L_2$ und L_1 NP-schwer ist, dann ist auch L_2 NP-schwer.

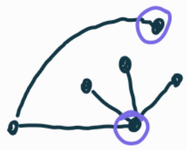
Def: • KNF: konjuktive Normalform;
Menge von Formeln, die nur aus bool'schen Ausdrücken mit "$\wedge$" und "$\vee$" bestehen.
$F \in KNF$, falls $F = F_1 \wedge F_2 \wedge \dots \wedge F_n$ wobei $F_i = (x_1 \vee x_2 \vee \dots \vee x_m)$.

Def: • k-Clique: Vollständiger Teilgraph mit k Knoten
↳ d.h. jeder Knoten innerhalb des Teilgraphen ist mit jeden anderen dieser k Knoten verbunden.



Def: · Knotenüberdeckung: (vertex-cover)

Knotenmenge minimaler Anzahl eines Graphen G, s.d. jede Kante von G an einem Ende mindestens einen dieser Knoten hat.



Mächtigkeit = 2

wichtige Sprachen

SAT $:= \{ \Phi \mid \Phi$ ist eine Formel in KNF$\}$

3SAT $:= \{ \Phi \mid \Phi \in$ KNF und jede Klause von Φ enthält höchstens 3 Variabeln$\}$

E3SAT $= \{ \Phi \mid \Phi \in$ KNF und jede Klause von Φ enthält genau 3 Variabeln$\}$

MonoSAT $:= \{ \Phi \mid \Phi \in$ KNF und Φ monoton (jede Klause enthält nur positive oder nur negierte Variabeln)$\}$

TRIPEL-SAT $:= \{ \Phi \mid \Phi \in$ KNF und Φ hat mindestens 3 erfüllende Bedingungen$\}$

CLIQUE $:= \{ (G,k) \mid G$ ist ein ungerichteter Graph, der eine k-Clique enthält$\}$

VC $:= \{ (G,k) \mid G$ ist ein ungerichteter Graph mit einer Knotenüberdeckung der Mächtigkeit höchstens $k \}$

Satz 6.9: (Satz von Cook)
SAT ist NP-vollständig

Lem 6.9: SAT $\leq_p$ CLIQUE

Lem 6.10: CLIQUE $\leq_p$ VC

Lem 6.11: SAT $\leq_p$ 3SAT

Aufg 2) Sei X eine Menge und $\mathcal{S} \subseteq \mathcal{P}(X)$ mit $\bigcup_{S \in \mathcal{S}} S = X$.

Eine Teilmenge $\mathcal{C} \subseteq \mathcal{S}$ ist eine Mengenüberdeckung (Set-Cover) von X, falls $\bigcup_{S \in \mathcal{C}} S = X$.

Das Set-Cover-Problem (SCP) ist def. als:

$$SCP := \{(X, \mathcal{S}, k) \mid X \text{ hat ein set-cover } \mathcal{C} \subseteq \mathcal{S} \text{ mit } |\mathcal{C}| \leq k\}$$

Zz: $VC \leq_p SCP$

Bmk: Wir werden die Lösung zur Aufgabe 2 sowie die Definitionen der wichtigen Sprachen nächste Woche gemeinsam anschauen.

Falls ihr für Serie 9 eine polynomielle Reduktion sehen wollt, empfehle ich euch, auf VIS bei den Finals (nicht endterms!) Aufgabe 5 von FS21 anzuschauen.

(Bei dieser Aufgabe geht es, wie in der Serie, auch um SAT-ähnliche Probleme.)