Universität Paderborn
Prof. Dr. Johannes Blömer

Gruppe 27:
Eli Kogan-Wang (7251030)
David Noah Stamm (7249709)
Bogdan Rerich (7248483)
Jan Schreiber(7253698)

Berechenbarkeit und Komplexität – WS 2022/2023

Heimübung 8
Abgabe: 12. Dezember 2022 – 13:00 Uhr
(Dieser Übungszettel besteht aus 4 Aufgaben mit insgesamt 24 Punkten)

Aufgabe 1 (k-Färbbarkeit in NP) (6 Punkte)
Ein ungerichteter Graphen $G = (V, E)$ heiSSt *k-färbbar*, wenn es eine Abbildung $c : V \to \{1, 2, \ldots, k\}$ gibt, sodass $c(u) \neq c(v)$ für alle $\{u, v\} \in E$. Zeigen Sie, dass die Sprache

$$\text{Color} = \{\langle G, k\rangle \mid G \text{ ist ein ungerichteter Graph, der eine } k\text{-Färbung besitzt}\}$$

in NP liegt. Konstruieren Sie hierfür eine NTM, welche die Sprache in polynomieller Zeit entscheidet, und beweisen Sie deren Korrektheit.

Lösung: Sei im folgenden N eine NTM
N bei Eingabe x $\in \{0,1\}*$

1. (Formcheck) Prüfe, ob x von der Form $\langle G, k\rangle$ ist, wobei G der Kodierung eines Graphen entspricht und k der einer natürlichen Zahl
2. Generiere nicht deterministisch Abbildungen $c : V \to \{1, 2, \ldots, k\}$
3. Prüfe $\forall$ $\{u, v\} \in E$, ob c(u) $\neq$ c(v), falls dies nicht der Fall ist verwerfe
4. Akzeptiere

Laufzeit von N

1. Überprüfen ist in P
2. Die Abbildung kann in polynomieller Zeit generiert werden, da jedem Knoten systematisch ein Element aus $\{1, 2, \ldots, k\}$ zugeordnet wird.
3. Dies dauert hat im Worst Case eine Laufzeit von $|V|^2$ also ist diese Polynomiel
4. Trivialerweise Polynomiel.

Korrektheit:

Behauptung: L(N) = Color.

1) Color $\subseteq$ L(N)

Angenommen x $\in$ Color
$\rightarrow$ Es existert ein k $\in \mathbb{N}$ s.d. x=$\langle G, k\rangle$ und G besitzt eine k-Färbung.
$\rightarrow$ Es existiert eine Abbildung $w: V \rightarrow \{1, 2, \ldots, k\}$ mit $w(u) \neq w(v)$ für alle $\{u, v\} \in E$.
$\rightarrow$ x wird in 1 nicht abgelehnt. Die Abbildung w wird in 2 generiert und die Bedingungen in 3 treten nach Voraussetzung ein, folglich wird x in 4 akzeptiert.
$\rightarrow$ Es existiert eine Berechnung in welcher x akzpetiert wird.
$\Rightarrow$ x $\in$ L(N)

2) L(N) $\subseteq$ Color

Abgenommen x $\notin$ Color
$\rightarrow$ x $\neq \langle G, k\rangle$, dann wird x in (1) abgelehnt oder x= $\langle G, k\rangle$, aber G ist nicht k färbbar.
$\rightarrow$ Es kann in 2 keine Abbildung $c: V \rightarrow \{1, 2, \ldots, k\}$ generiert werden für die die Bedingungen aus 3) erfüllt werden können.
$\rightarrow$ x wird in jeder Berechnung abgelehnt.
$\Rightarrow$ x $\notin$ L(N).
$\Rightarrow$ Color=L(N)
$\Rightarrow$ Color $\in$ NP $\square$

Aufgabe 2 (P und Implikation) (6 Punkte)
Eine aussagenlogische Formel ϕ ist in *disjunktiver Normalform* (DNF), wenn sie von der Form $\phi = \bigvee_{i=1}^{k} \bigwedge_{j=1}^{m_i} x_{ij}$ ist, wobei x_{ij} Literale sind. Betrachten Sie die folgende Sprache

$$\textsc{DisSat} = \{\langle\phi\rangle \mid \phi \text{ ist eine erfüllbare Formel in disjunktiver Normalform}\} .$$

1. Zeigen Sie, dass DisSat in P liegt.
2. Jede 3KNF-Formel, das heiSSt KNF-Formel deren Klauseln aus 3 Literalen bestehen, kann in eine logisch äquivalente DNF-Formel umgeformt werden (wie Sie wahrscheinlich aus der Vorlesung Modellierung wissen). Warum implizieren dieser Fakt und a) *nicht* notwendigerweise, dass auch 3SAT in P liegt?

Lösung:

1. Wir zeigen eine Turing Maschine M, die die Sprache DisSat in polynomieller Zeit entscheidet.

   M bei eingabe w:

   a) Wenn w nicht von der Form $\langle\phi\rangle$ für eine Formel ϕ in DNF ist, wird w abgelehnt.
   b) $w = \langle\phi\rangle$ für eine Formel $\phi = \bigvee_{i=1}^{k} K_i$ in DNF. Wobei $K_i = \bigwedge_{j=1}^{m_i} x_{ij}$ ein Konjunktionsterm ist. Für jeden Konjunktionsterm K_i wird folgendes getan:

      i. Prüfe für alle Literale x_{ij} in K_i, ob $\neg x_{ij}$ nicht in ϕ vorkommt. Wenn es nicht vorkommt, wird w akzeptiert.

ii. Wenn x_{ij} und $\neg x_{ij}$ in ϕ vorkommt, wird der nächste Konjunktionsterm K_{i+1} betrachtet.

c) Wenn alle Konjunktionsterme betrachtet wurden, wird w abgelehnt.

Die Laufzeit von M ist polynomiel, da für jede Formel ϕ in DNF nur die Konjunktionsterme in ϕ betrachtet werden müssen.

Die Betrachtung der Konjunktionsterme ist polynomiel, da für jeden Konjunktionsterm K_i nur die Literale in K_i betrachtet werden müssen, und naiverweise nur jedes Paar von Literalen in K_i betrachtet werden muss (quadratische Vergleiche).

Wenn M die Formel ϕ in DNF akzeptiert, dann ist ϕ erfüllbar, mit einer Interpretation I die alle Literale des K_i erfüllt, in dem M akzeptiert.

Wenn M nicht akzeptiert, dann ist entweder:

a) Die Eingabe w nicht von der Form $\langle \phi \rangle$ für eine Formel ϕ in DNF.

b) ϕ nicht erfüllbar, da alle Konjunktionsterme wiederspruchsvoll sind.

$\Rightarrow$ DisSat ist in P.

2. Dieser Fakt impliziert nicht notwendigerweise, dass auch 3SAT in P liegt, da die uns bekannte Konversion nicht polynomiel ist.

   Insbesondere wandelt der uns bekannte Algorithmus (Distributivgesetz anwenden) eine 3KNF-Formel ϕ in eine DNF-Formel ϕ' um, die ϕ logisch äquivalent ist, aber $O(3^n)$ (mit n der Anzahl der Klauseln in ϕ) Terme hat. Bei einer Formel ϕ mit n Klauseln ist eine der Terme dann $O(n)$ lang und somit ist die Laufzeit der Reduktion auf ϕ' nicht polynomiel.

   Damit ist keine polynomielle Reduktion von 3SAT auf DisSat gegeben, und somit ist 3SAT nicht notwendigerweise in P.

Aufgabe 3 (SetCover) (6 Punkte)

Zeigen Sie, dass die folgende Sprache in NP liegt:

$$\text{SetCover} = \left\{ \langle A, S_1, \ldots, S_n, k \rangle \;\middle|\; \begin{array}{l} S_1, \ldots, S_n \subseteq A \land k \in \mathbb{N} \land \exists I \subseteq \{1, \ldots, n\} \\ \text{mit } |I| \leq k \text{ und } \bigcup\limits_{i \in I} S_i = A \end{array} \right\}$$

Lösung:

o.B.d.A alle hier betrachten Mengen sind endlich.

Sei V im Folgenden eine DTM

V bei Eingabe $\in \{0,1\}^*$

1. (Formcheck) Überprüfe, ob x = $\langle A, S_1, \ldots, S_n, k, I \rangle$ mit $S_1, \ldots, S_n \subseteq A \land k \in \mathbb{N} \land I \subseteq \{1, \ldots, n\}$mit $|I| \leq k$

2. Prüfe, ob $\bigcup_{i\in I} S_i = A$, falls nicht verwerfe

3. Akzeptiere

Behauptung: V ist ein polynomieller Verifzierer der Sprache SetCover.

Korrektheit:

Angenommen x $\in$ SetCover.
Also ist x $= \langle A, S_1, \ldots, S_n, k\rangle$ mit $S_1, \ldots, S_n \subseteq A \wedge k \in \mathbb{N}$ und es existiert eine Indexmenge I s.d. $|I| \leq k$ und $\bigcup_{i\in I} S_i = A$
Mit I als Zertifikat wird dann dem Entsprechend V in Schritt 1 und 2 die Eingabe nicht verwerfen und somit in 3 akzeptiert.

Angenommen x $\notin$ SetCover.
Also ist x entweder nicht von der richtigen Form, wird also 1) abgelehnt oder es existiert keine Indexmenge, welche die geforderten Bedingungen nicht erfüllt.
Also lehnt auch V bei jeder Paarung von x und einer Indexmenge nach 1 und 2 ab, da entweder Bedingungen aus dem Formcheck nicht erfüllt sind oder $\bigcup_{i\in I} S_i \neq A$, da x sonst $\in$ SetCover.
$\Rightarrow$ V ist ein Verfizierer von SetCover
Laufzeit von V:

1. Hat polynomielle Laufzeit, da das überprüfen der Mengen Inklusionen in $\mathcal{O}(|A|)$ möglich ist. Ob k in **N** liegt ist auch trivialerweise in polynmieller Zeit möglich, genauso wie

2. Dies ist in polynomieller Laufzeit umsetztbar, da die Vereinigung von Mengen in polynomieller Zeit abhängig von der groSSe der Mengen umgesetzt werden kann und der Vergleich in $\mathcal{O}(|A|)$ berechnet werden kann.

3. trivialerweise polynomiell

Polynomialität von I:
Da I $\subseteq \{1, \ldots, n\}$. Kann dieses maximal n Elemente enthalten und kann somit in $\mathcal{O}(|A|)$ berechnet werden.
$\Rightarrow$ V ist polynomieller Verifizierer von SetCover $\Rightarrow$ SetCover $\in NP$

Aufgabe 4 (Implikationen) (6 Punkte)
Seien A, B, C, D beliebige Sprachen. Nehmen Sie zusätzlich folgende Beziehungen an

- Es gibt eine Reduktion von A zu B, also $A \leq B$.
- Es gibt eine Reduktion von B zu C, also $B \leq C$.
- Es gibt eine Reduktion von D zu C, also $D \leq C$.

Entscheiden und Begründen Sie für jede der folgenden logischen Aussagen kurz, ob sie

(i) immer wahr ist, also für alle Wahlen von Sprachen erfüllt ist.

(ii) möglicherweise wahr ist, also sowohl eine erfüllende als auch eine widersprechende Wahl von Sprachen existiert.

(iii) immer falsch ist, also für keine Wahl von Sprachen erfüllt ist.

Die Aussagen lauten:

a) A is rekursiv aufzählbar aber nicht entscheidbar, und C ist entscheidbar.

b) A ist nicht entscheidbar und D is nicht rekursiv aufzählbar.

c) Wenn C entscheidbar ist, dann ist das Komplement von D entscheidbar.

d) Wenn C rekursiv aufzählbar ist, dann ist $B \cap D$ rekursiv aufzählbar.

Lösung:

a) (iii), da Reduktionen Transitiv sind $\Rightarrow$ A $\leq$ C und C entscheidbar $\xrightarrow{2.8.2}$ A entscheidbar.

b) A nicht entscheidbar und A $\leq$ C $\xrightarrow{2.9.1}$ C ist nicht entscheidbar $\iff$ C oder C^c ist nicht rekursiv aufzählbar.
D nicht rekursiv aufzählbar $\xrightarrow{2.9.1}$ C ist nicht rekursiv aufzählbar. $\Rightarrow$ (ii) ist richtig.

c) (i), da D $\leq$ C und C entscheidbar $\xrightarrow{2.8.2}$ D ist entscheidbar $\xrightarrow{2.5.1}$ D^c ist entscheidbar.

d) (i), da D $\leq$ C, B $\leq$ C und C rekursiv aufzählbar $\xrightarrow{2.8.2}$ D,B sind rekursiv aufzählbar $\xrightarrow{2.5.2}$ $B \cap D$ ist rekursiv aufzählbar.

Angenommen C ist rekursiv aufzählbar + D $\leq$ C $\rightarrow$ D ist aufzählbar. Angenommen C ist nicht rekursiv aufzählbar + D $\leq$