Prof. Dr. Jasmin Blanchette
Jannis Limperg
Lydia Kondylidou

Ludwig-Maximilians-Universität München
Institut für Informatik
Keine Abgabe

Lösungsvorschlag zur Übung 13 zur Vorlesung

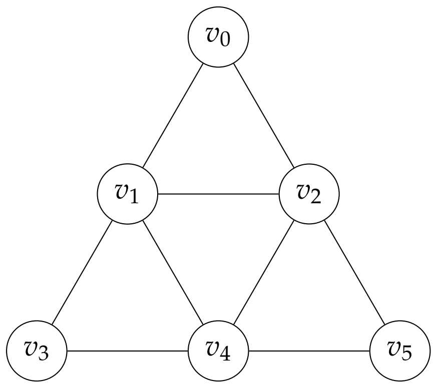
Theoretische Informatik für Medieninformatiker

Dieses Übungsblatt kann aus Zeitgründen nicht mehr abgegeben werden und wird somit auch nicht korrigiert.

TIMI13-1 ***$\mathcal{NP}$-Vollständigkeit von Graphenproblemen*** (0 Punkte)

Das Independent-Clique-Problem (ICP) beantwortet für einen Graphen G und eine natürliche Zahl n die Frage, ob es in G eine Clique C der Größe n und ein Independent Set I der Größe n gibt, sodass $|C \cap I| = 1$ gilt.

a) Der Graph G sei:



Beantworten Sie für alle $n \in \mathbb{N}$ die Frage, ob $(G, n) \in$ ICP gilt. Begründen Sie Ihre Antworten.

LÖSUNGSVORSCHLAG:

Als Bemerkung: Eine Clique und ein Independent Set können sowieso nur maximal einen Knoten gemeinsam haben.

- Für $n = 1$: Jeder der 6 Knoten allein ist eine Clique und ein Independent Set. Daher gilt $(G, 1) \in$ ICP.
- Für $n = 2$: Wähle zwei verbundene Knoten für die Clique, einen der beiden Knoten und einen nicht mit ihm verbundenen Knoten für das Independent Set. Diese gibt es, z.B. $C = \{v_0, v_1\}$, $I = \{v_0, v_3\}$, also ist $(G, 2) \in$ ICP.

- Für $n = 3$: Wähle $C = \{v_0, v_1, v_2\}$ und $I = \{v_0, v_3, v_5\}$, somit $(G, 3) \in$ ICP.
- Für $n \geq 4$: Der Graph hat keine Clique der Größe 4 oder größer. Daher ist $(G, n) \notin$ ICP für alle $n \geq 4$.

b) Zeigen Sie, dass ICP $\mathcal{NP}$-vollständig ist.

LÖSUNGSVORSCHLAG:

ICP $\in \mathcal{NP}$: Rate nichtdeterministisch eine Knotenmenge C und eine Knotenmenge I von jeweils n Knoten. Verifiziere anschließend, ob C eine Clique, I ein Independent Set, und C und I genau einen gemeinsamen Knoten haben. Wenn ja, akzeptiere, wenn nein, verwirf. Das Verifizieren ist in deterministischer Polynomialzeit möglich.

ICP ist $\mathcal{NP}$-schwer: Wir reduzieren CLIQUE auf ICP mit einer Polynomialzeitreduktion. Da CLIQUE $\mathcal{NP}$-vollständig ist, ist ICP damit $\mathcal{NP}$-schwer.

Sei w ein Wort. Wenn $w \neq code(V, E, k)$, dann sei $f(w) = w$. Anderenfalls sei $f(w) = code(V', E, k)$, wobei $V' = V \cup \{v_1, \ldots, v_{k-1}\}$, sodass $\forall i \in \{1, \ldots, k-1\}$: $v_i \notin V$ (d.h. $v_1, \ldots, v_{k-1}$ sind neue Knoten).

Damit ist f total und in Polynomialzeit berechenbar. Wir zeigen die Korrektheit von f:

- Wenn w keine gültige Eingabe für CLIQUE ist, dann ist $f(w)$ auch keine gültige Eingabe für ICP.
- Wenn $w = code(V, E, k)$ und (V, E, k) hat eine k-Clique, dann hat (V', E, k) ebenfalls eine k-Clique. Sei u ein beliebiger Knoten dieser Clique. Dann ist $\{u, v_1, \ldots, v_{k-1}\}$ ein Independent Set der Größe k, somit $(V', E, k) \in$ ICP.
- Wenn $w = code(V, E, k)$ und (V, E, k) hat keine k-Clique, dann hat (V', E, k) ebenfalls keine k-Clique, somit $(V', E, k) \notin$ ICP.

TIMI13-2 ***SAT-Varianten in $\mathcal{P}$ und $\mathcal{NP}$*** (0 Punkte)

a) Sei UNSAT = $\{code(F) \mid F$ ist eine widersprüchliche Formel$\}$. Nehmen Sie an, dass UNSAT in $\mathcal{NP}$ ist.

Wir betrachten folgende Reduktionsfunktion von SAT auf UNSAT: Teste alle möglichen Variablenbelegungen der Formel. Wenn eine erfüllende Variablenbelegung gefunden wurde, gib $x \wedge \neg x$ zurück, ansonsten x.

Ist diese Reduktionsfunktion geeignet, um zu zeigen, dass UNSAT $\mathcal{NP}$-vollständig ist? Begründen Sie Ihre Antwort.

LÖSUNGSVORSCHLAG:

Nein, denn diese Reduktionsfunktion hat keine polynomielle, sondern exponentielle Laufzeit, da sie exponentiell viele Schritte braucht um alle möglichen Variablenbelegungen zu testen.

b) Sei 3mal-3SAT = $\left\{ code(F) \;\middle|\; \begin{array}{l} F \text{ ist eine 3-CNF und hat mindestens} \\ \text{3 verschiedene erfüllende Belegungen} \end{array} \right\}$.

Zeigen Sie, dass 3mal-3SAT $\mathcal{NP}$-vollständig ist, wobei Sie für den Nachweis der $\mathcal{NP}$-Schwere eine Reduktion von 3-CNF-SAT auf 3mal-3SAT durchführen.

Hinweis: Was könnten Sie in der benötigten Reduktionsfunktion f hinzufügen, um aus einer erfüllenden Belegung 3 erfüllende Belegungen zu erzeugen?

LÖSUNGSVORSCHLAG:

Für die $\mathcal{NP}$-Vollständigkeit von 3mal-3SAT ist zu zeigen:

- 3mal-3SAT ist in $\mathcal{NP}$: Betrachte die folgende NTM M:
  - M berechnet aus $code(F)$ die Menge der in F vorkommenden aussagenlogischen Variablen.
  - M rät nichtdeterministisch ein Tupel von Belegungen (I_1, I_2, I_3) der Variablen von F. Das geht in nichtdeterministischer Polynomialzeit, indem man I_1 für jede Variable x in die Möglichkeiten $x = 0$ und $x = 1$ nichtdeterministisch verzweigt, für I_2 nochmal dasselbe durchführt, und für I_3 auch nochmal dasselbe.
  - Falls $I_1 = I_2 \vee I_1 = I_3 \vee I_2 = I_3$ verwirf diese nichtdeterministische Berechnung (der Test kann in Polynomialzeit durchgeführt werden)
  - Sonst prüfe, ob $I_1(F) = 1$, $I_2(F) = 1$ und $I_3(F) = 1$. Wenn ja, dann akzeptiere, anderenfalls verwirf diese nichtdeterministische Berechnung (die Tests können dabei in Polynomialzeit durchgeführt werden)

  M entscheidet 3mal-3SAT (denn es werden 3 verschiedene Lösungen gesucht) und M benötigt nur polynomiell viele Schritte (denn es handelt sich um eine Hintereinanderausführung von endlich vielen jeweils polynomiell langen Abfolgen von Schritten). Somit ist 3mal-3SAT in $\mathcal{NP}$.
- 3mal-3SAT ist $\mathcal{NP}$-schwer. Wir reduzieren 3-CNF-SAT $\leq_p$ 3mal-3SAT. Dazu definieren wir zunächst die Reduktionsfunktion f:

$$f(w) = \begin{cases} code(F \wedge (x \vee y \vee z)) & \text{falls } w = code(F) \text{ für eine Formel } F \\ w & \text{sonst} \end{cases}$$

Dabei sind x, y und z neue Variablen. Die Funktion f ist total und in Polynomialzeit berechenbar.

Wir zeigen: F ist erfüllbar g.d.w. $f(F)$ dreimal erfüllbar ist:

- „$\Rightarrow$“: Wenn $I(F) = 1$, dann machen $I_1 = I \cup \{x \mapsto 1, y \mapsto 0, z \mapsto 0\}$, $I_2 = I \cup \{x \mapsto 0, y \mapsto 1, z \mapsto 0\}$ und $I_3 = I \cup \{x \mapsto 0, y \mapsto 0, z \mapsto 1\}$ alle F wahr.
- „$\Leftarrow$“: Wenn $I_i(F \wedge (x \vee y \vee z)) = 1$ für $i = 1, 2, 3$, dann sei $I(v) = I_1(v)$ für alle Variablen v aus F. Dann gilt $I(F) = 1$.

Damit folgt 3-CNF-SAT $\leq_p$ 3mal-3SAT.

c) Sei Pos-3-SAT = $\left\{ code(F) \;\middle|\; \begin{array}{l} F \text{ ist eine 3-CNF, in der ausschließlich} \\ \text{positive Literale vorkommen} \end{array} \right\}$.

i) Liegt Pos-3-SAT in $\mathcal{P}$?

LÖSUNGSVORSCHLAG:

Eine Pos-3-SAT-Formel ist bereits erfüllbar, wenn jede Klausel mindestens eine Variable enthält: Wenn man alle Variablen auf 1 setzt, sind alle Klauseln erfüllt. Das ist in Polynomialzeit überprüfbar.

Ferner muss noch überprüft werden, ob das Wort eine gültige Kodierung einer Formel ist; auch das ist in Polynomialzeit überprüfbar.

Also liegt Pos-3-SAT in $\mathcal{P}$.

ii) Liegt Pos-3-SAT in $\mathcal{NP}$?

LÖSUNGSVORSCHLAG: $\mathcal{P} \subseteq \mathcal{NP}$, somit liegt Pos-3-SAT auch in $\mathcal{NP}$.

Begründen Sie jeweils Ihre Antworten.