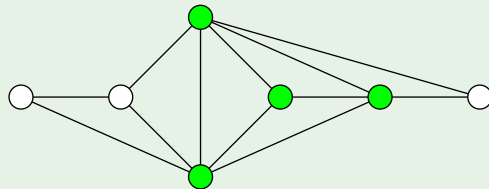


Cliquen

Definition

Es sei $G = (V, E)$ ein ungerichteter Graph. Eine Eckenmenge $W \subseteq V$ heie eine *Clique*, wenn fr alle Ecken $u, v \in W$ gilt: $[u, v] \in E$. (Die Ecken aus W bilden einen vollstndigen Untergraphen.)

Beispiel



Die markierten Ecken bilden eine Clique der Gre 4.



Clique-Problem

Varianten des Clique-Problems

- Entscheidungsproblem
- Optimierungsproblem
- Berechnungsproblem

Wir haben festgestellt:

Ist eine der Varianten in polynomieller Zeit berechenbar, dann auch jede der anderen.



Clique-Problem (Fortsetzung)

Frage:

Gibt es einen effizienten Algorithmus fr das Clique-Problem?

Vermutung:

Nein.

Aber: Das Clique-Problem (Entscheidungs-Variante) ist in NP.
D. h. es gibt einen nichtdet. Algorithmus, der in Polynomialzeit verifiziert, dass G eine Clique der Gre k hat.

Bemerkung

Das ist ein Beispiel von vielen wichtigen Problemen, fr die man eine Lsung effizient verifizieren kann, man aber nicht wei, wie man eine Lsung effizient findet.



Polynomialzeitreduktionen

Definition

Es seien $L, L' \in \Sigma^*$. Wir nennen L *polynomiell leichter* als L' bzw. L *polynomiell reduzierbar* auf L' , wenn es eine Funktion $f : \Sigma^* \rightarrow \Sigma^*$ gibt mit

- 1 $\forall w \in \Sigma^* : w \in L \Leftrightarrow f(w) \in L'$,
- 2 f ist in polynomieller Zeit berechenbar.

Wir schreiben $L \preceq_P L'$.

Beispiel

Jede Variante des Clique-Problems ist polynomiell reduzierbar auf jede Variante des Clique-Problems.



Polynomialzeitreduktionen

Lemma

Es seien $L, L' \in \Sigma^*$. Dann gilt

$$L \preceq_P L' \text{ und } L' \in P \Rightarrow L \in P.$$

Die Aussage ist auch richtig, wenn man P durch NP ersetzt.

Beweis.

- $L \preceq_P L'$: Es gibt TM F , die $f: \Sigma^* \rightarrow \Sigma^*$ berechnet mit $w \in L \Leftrightarrow f(w) \in L'$, Laufzeit $O(n^k)$.
- $L' \in P$: Es gibt (det.) TM M' , die L' entscheidet, Laufzeit $O(n^l)$.
- Wir bauen eine (det.) TM M für L (Eingabe w):
 - 1) Berechne $f(w)$ mit F .
 - 2) Teste mit M' ob $f(w) \in L'$ ist.
- M entscheidet L , Laufzeit: $O(n^k) + O(|f(w)|^l) \in O(n^{kl})$ □



NP-schwer – NP-vollständig

Definition

- 1 Eine Sprache S heie *NP-schwer*, wenn fr alle $L \in NP$ gilt $L \preceq_P S$.
- 2 Eine Sprache S heie *NP-vollstndig*, wenn
 - 1 S NP-schwer ist und
 - 2 $S \in NP$ gilt.

Satz

Es sei $S \in \Sigma^*$. Dann gilt

$$S \text{ NP-schwer/vollstndig und } S \in P \Rightarrow P = NP.$$

Beweis.

- 1 $P \subseteq NP$: ✓
 - 2 $NP \subseteq P$:
 - ▶ $L \in NP$ beliebig
 - ▶ S NP-schwer $\Rightarrow L \preceq_P S$
 - ▶ $S \in P \Rightarrow L \in P$ (vgl. 1. Lemma)
-



Polynomialzeitreduktionen

Lemma (Transitivitt)

Es seien $L, L', L'' \in \Sigma^*$. Dann gilt

$$L \preceq_P L' \text{ und } L' \preceq_P L'' \Rightarrow L \preceq_P L''.$$

Beweis.

Vergleiche bungsblatt 8, Aufgabe 2 und voriges Lemma. □



NP-schwer – NP-vollstndig

Andere Formulierung des Satzes

$$P \neq NP \text{ und } S \text{ NP-vollstndig} \Rightarrow S \notin P$$

Interpretation

Wenn man von einer Sprache S zeigt, dass sie NP-vollstndig ist, dann ist dies auf Grund der Vermutung $P \neq NP$ ein starker Grund zur Annahme, dass $S \notin P$, dass also S nicht effizient lsbar ist.



NP-schwer – NP-vollständig

Wie zeigt man, dass eine Sprache NP-schwer ist?

- direkt oder
- mit dem folgenden Lemma:

Lemma

Es seien $L, S \in \Sigma^*$. Dann gilt

L NP-schwer und $L \preceq_P S \Rightarrow S$ NP-schwer.

Beweis.

- $L' \in \text{NP}$ beliebig
- L NP-schwer $\Rightarrow L' \preceq_P L$
- zusammen mit $L \preceq_P S \Rightarrow L' \preceq_P S$ (Transitivität von $\preceq_P$) □



NP-vollständige Probleme

Erinnerung / Zusammenfassung

Eine Sprache S ist NP-vollständig, wenn

- 1 $S \in \text{NP}$ und
- 2 wenn für eine beliebige NP-vollständige Sprache L gilt $L \preceq_P S$.

Gibt es NP-vollständige Probleme?

Ja.



Wer wird Millionär?

1 000 000 \$-Frage

- Welche der beiden Aussagen ist richtig?
a) $P = \text{NP}$ b) $P \neq \text{NP}$
- „Zuschauertip“: b)
- Aber: Mit Beweis!

Clay Mathematics Institute – Millennium Problems

- Sieben Probleme – darunter P vs. NP –, jeweils 1 000 000 \$ für die Lösung
- <http://www.claymath.org/millennium/>



P-UNIV

Definition

$P\text{-UNIV} =$

$\{\text{cod}(M) \$ w \$ 1^m \mid \text{nichtdet.TM } M \text{ akzeptiert } w \text{ in } \leq m \text{ Schritten}\}$

Behauptung

$P\text{-UNIV}$ ist NP-vollständig.

