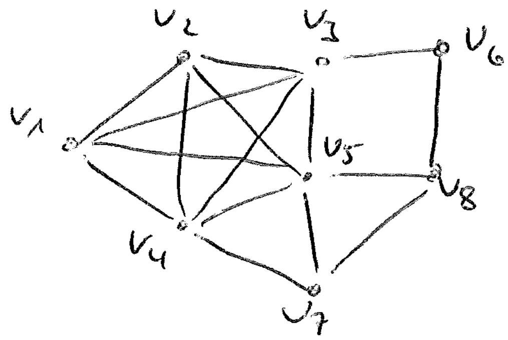
Problem "Clique"

Gegeben: Graph $G=(V,E)$, $k\in\mathbb{N}$

Frage: Existiert $S\subseteq V$ mit $|S|=k$, sodass für je $u,v\in S$ gilt $\{u,v\}\in E$.

Bsp:



Graph enthält eine Clique der Größe 5 $(v_1, v_2, v_3, v_4, v_5)$.

Zeige: Clique ist NP-vollständig

1. Clique ist in NP.

Beweis:

Produziere ein Zertifikat $\{0,1\}^n$ für n Knoten.

Ein Verifizierer überprüft nun zwei Punkte für ein Zertifikat:

a) Die Anzahl 1en beträgt genau k

b) Für jedes Paar i,j, für welche die i-te und j-te Ziffer des Zertifikats eine 1 besitzt, prüfe ob $\{v_i, v_j\}\in E$.

Es wurden also k Knoten ausgewählt, die paarweise verbunden sind.
Gibt es keine Clique der Größe k, schlägt die Überprüfung immer fehl.

~~2. Clique ist~~ 2. Clique ist NP-schwer.

Welches Problem nutzt man für eine Reduktion?

$\underbrace{\text{3SAT, Hamiltonkreis,}}_{\text{VL}}$ $\underbrace{\text{Vertex Cover, Independent Set}}_{\text{Hausaufgaben}}$

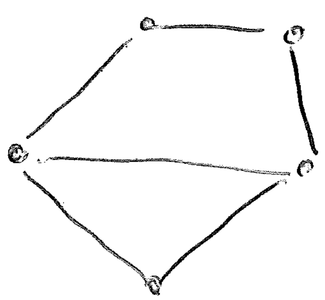
Vergleiche die Probleme: Independent Set ist sehr ähnlich!

Clique $\longleftrightarrow$ Independent Set

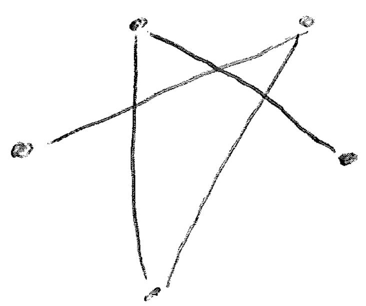
Alle verbunden $\Longleftrightarrow$ gar nicht verbunden.

Idee: Betrachte Graphen $\overline{G} = (V, \overline{E})$ mit $\overline{E} = \binom{V}{2} \setminus E$, d.h. wir nehmen alle Kanten auf, die nicht in G existierten, und lassen alle aus, die in G existieren.

Bsp: G



$\overline{G}$



Klar, diese Reduktion lässt sich in Polyzeit durchführen.

Bleibt Korrektheit zu zeigen.

Zu zeigen:

G enthält Clique der Größe k, gdw. $\overline{G}$ enthält unabhängige Menge der Größe k.

„$\Longrightarrow$" Betrachte Clique S in G der Größe k. Da alle untereinander verbunden sind, existieren unter S in $\overline{G}$ gar keine Kanten, d.h. S ist ein UM in $\overline{G}$.

„$\Longleftarrow$" analog nur mit Start in $\overline{G}$.

Problem "Set Cover"

Potenzmenge von U, d.h. S ist eine Menge von Teilmengen.

Gegeben: $U = \{1, \dots, n\}$ und $S \subseteq P(U)$,

sowie $k \in \mathbb{N}$

Frage: Existiert $C \subseteq S$ mit $|C| = k$ und

$$\bigcup_{s \in C} s = U \ ?$$

Beispiel:

$U = \{1, 2, 3, 4, 5\}$

$S = \{\{1, 2, 3\}, \{2, 4\}, \{3, 4\}, \{4, 5\}\}$

$k = 2$

$\Rightarrow C = \{\{1, 2, 3\}, \{4, 5\}\}$

Zeige: Set Cover ist NP-vollständig

1. Set Cover ist in NP.

Auch hier besteht das Zertifikat aus einem einfachen Binärstring der Länge $|S|$, welcher angibt, welche Mengen aus S im Set Cover liegen sollen.

Für ein Zertifikat prüfe folgendes:

1. Anzahl 1en beträgt ~~≤~~ k.
2. Liste alle Elemente aller $s_i \in S$ auf, für welche an i-ter Stelle im ~~B~~ Zertifikat eine 1 steht.
3. Prüfe, ob jedes Element in U in dieser Liste auftaucht.

Laufzeit:

Die Liste beinhaltet maximal $O(k \cdot n) \subset O(|S| \cdot n)$ viele Elemente. Die Überprüfung, ob jedes Element abgedeckt ist kann damit in $O(|S| n^2)$ Zeit erfolgen. Das ist polynomiell im Input!

Bemerkung: Das geht schneller, uns reicht aber polyzeit!

Korrektheit:

Wenn es ein Set Cover C der Größe k gibt, dann setzen wir an der i-ten Stelle des Zertifikats eine 1, ~~Dann~~ wenn $s_i \in C$. Damit hat das Zertifikat k 1en und die Überprüfung muss auch gelingen.

Andersherum: Haben wir ein Zertifikat, dessen Überprüfung gelingt, dann ist dies nach Konstruktion auch ein Set Cover.

6

2. Set Cover ist NP-schwer

Zeige: Vertex Cover $\leq_m^{poly}$ Set Cover

Man sieht recht schnell:

Die Kanten E entsprechen dem Universum und die an einem Knoten v anliegenden Kanten entsprechen einer Menge in S.

Sei $\delta(u) = \{ \{u,v\} \mid \{u,v\} \in E, v \in V \}$

Dann ist $S = \{ \delta(u) \mid u \in V \}$

und $U = E$

Diese Reduktion lässt sich in Polyzeit durchführen:
U berechnen müssen wir nicht, es bleibt E.
S zu berechnen dauert für jeden Knoten maximal $O(n)$ Zeit. Insgesamt also $O(n^2)$ für n Knoten.

Claim

G hat ein Vertex Cover der Größe k, gdw. die Set Cover Instanz (U, S) ein Set Cover der Größe k besitzt.

"$\Rightarrow$" Angenommen, es gibt ein VC der Größe k mit Knoten $\{v_1, \dots, v_k\}$. Dann ist

$$\begin{array}{ccc} \bigcup_{i=1}^{k} \delta(v_i) & = & E \\ \| & & \| \\ \bigcup_{i=1}^{k} S_i & = & U \end{array} \Rightarrow \text{Es ex. ein Set Cover der Größe } k$$

"$\Leftarrow$" ~~Es gibt ein Set Cover $\{S_1, \dots, S_k\}$ mit~~

Analog von unten nach oben, da es äquivalenzaussagen sind.

Problem "LP" (Linear Program)

Gegeben: • Kostenvektor $c \in \mathbb{R}^n$
• Koeffizientenmatrix $A \in \mathbb{R}^{m \times n}$
• Konstantenvektor $b \in \mathbb{R}^m$
• Variablen $(x_1, \dots, x_n) = \vec{x}$
• Zahl $k \in \mathbb{R}$

Frage: Existiert ein $\vec{x} \geq 0$ mit

$$A\vec{x} \leq b$$
$$c^T\vec{x} \geq k \quad ?$$

Man kann zeigen: $LP \in P$.

Zeige: LP ist P-schwer (bzgl logspace Reduktionen)

Idee: $CVP \leq_m^{\log} LP$

D.h. wir müssen aus den Zuweisungen P_i in CVP Bedingungen der Form $Ax \leq b$ für LP konstruieren.

Dazu: Für jede Zuweisung P_i erzeugen wir eine Variable x_i mit $0 \leq x_i \leq 1$

Betrachte nun die verschiedenen Zuweisungen

$P_i = 0 \quad \Rightarrow \quad x_i \leq 0$ bzw $x_i = 0$

$P_i = 1 \quad \Rightarrow \quad x_i \geq 1 \quad (\Leftrightarrow -x_i \leq -1)$

bzw $x_i = 1$

$P_i = \neg P_s \quad \Rightarrow \quad x_i = 1 - x_s$

$P_i = P_s \wedge P_t \quad \Rightarrow \quad x_i \leq x_s \;,\; x_i \leq x_t$

$x_i \geq x_s + x_t - 1$

$P_i = P_s \vee P_t \quad \Rightarrow \quad x_i \geq x_s \;,\; x_i \geq x_t$

$x_i \leq x_s + x_t$

Man kann recht schnell verifizieren, dass alle x_i entweder 0 oder 1 sein müssen.

Als letzte Bedingung können wir

$x_\ell = 1$ erzeugen.

Damit:

CVP ~~ka~~ gibt 1 heraus, gdw. das LP eine Lösung besitzt.

Laufzeit und Korrektheit selbst.