Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Übungsblatt 11

Aufgabe 1. Zeigen Sie, dass CLIQUE **NP**-vollständig ist.

Lösung. Seien $K_1, \ldots, K_m$ mit $m \geq 1$ die Klauseln der 3-SAT-Instanz und seien $x_1, \ldots, x_n$ die Variablen, die in ihr vorkommen. Sei $K_i = \{K_{i,1}, K_{i,2}, K_{i,3}\}$ die i'te Klausel, d.h. $K_{i,1}, K_{i,2}, K_{i,3} \in \{x_1, \neg x_1, \ldots, x_n, \neg x_n\}$. Wir definieren den folgenden ungerichteten Graph $G = (V, E)$ mit

$$V = \{v_{i,j} \mid 1 \leq i \leq m, 1 \leq j \leq 3\},$$
$$E = \{\{v_{i,j}, v_{i',j'}\} \mid 1 \leq i, i' \leq m, 1 \leq j, j' \leq 3, i \neq i', K_{i,j} \neq \overline{K_{i',j'}}\}.$$

Dieser Graph kann in Polynomialzeit konstruiert werden. Um zu entscheiden, ob die 3-SAT-Instanz erfüllbar ist, fragen wir, ob G eine Clique der Größe mindestens m enthält.
Enthält G eine Clique C der Größe mindestens m, muss diese mindestens ein Literal jeder Klausel enthalten, d.h. zu jedem $1 \leq i \leq m$ gibt es ein j mit $v_{i,j} \in C$. Die Clique kann nach Konstruktion keine sich widersprechenden Literale enthalten, d.h. für $v_{i,j} \neq v_{i',j'} \in C$ kann nicht $K_{i,j} = \overline{K_{i',j'}}$ gelten. Die Variablen belegen wir dann wie folgt: Wenn das Literal positiv ist, also $K_{i,j} = x$ für ein $x \in \{x_1, \ldots, x_n\}$, so wählen wir $\mathcal{B}(x) = 1$. Wenn das Literal negativ ist, also $K_{i,j} = \neg x$, wählen wir $\mathcal{B}(x) = 0$. Alle übrigen Variablen können beliebig belegt werden.
Wenn wir eine erfüllende Belegung $\mathcal{B}$ haben, finden wir eine Clique C der Größe mindestens m wie folgt: Wir fügen zu jedem $1 \leq i \leq m$ genau ein $v_{i,j}$ zu C hinzu mit $\mathcal{B}(K_{i,j}) = 1$, d.h. wir wählen aus jeder Klausel genau ein Literal aus, das unter $\mathcal{B}$ wahr ist. Diese Literale existieren, da $\mathcal{B}$ eine erfüllende Belegung ist. Sie sind außerdem alle miteinander verbunden, da sie in verschiedenen Klauseln liegen und sich nicht widersprechen können. C ist also eine Clique der Größe m.
Damit haben wir gezeigt, dass CLIQUE **NP**-schwer ist. Um zu zeigen, dass CLIQUE überhaupt in **NP** liegt, können wir bei einem gegebenen Graph (V, E) und einer Zahl m nichtdeterministisch eine Teilmenge von V der Größe m wählen. Zu testen, ob all diese Knoten paarweise miteinander verbunden sind, geht leicht in Polynomialzeit.

Aufgabe 2. Zeigen Sie, dass das Wortproblem für kontextfreie Grammatiken **P**-vollständig ist.

Lösung. Wir wollen MCVP auf das Wortproblem für KFGs reduzieren. Die Idee ist hierbei Folgende: Sei $C = (V, E, s)$ ein Circuit, wobei $v_o \in V$ die Ausgabe ist. Sei $w \in \{0,1\}^*$ das Blattform von C (die Labels s von links nach rechts). Wir konstruieren eine Grammatik $G = (\{0,1\}, N, S, P)$ mit $w \in L(G)$ genau dann, wenn C sich zu 1 auswertet. Dies funktioniert folgendermaßen: Jeder Knoten $v \in V$ erhält zwei Nichtterminale F_v und T_v. Hierbei erkennt F_v alle Blattwörter, unter denen sich der Teilcircuit unter v zu 0 auswertet. Entsprechend erkennt T_v alle Blattwörter, unter denen sich der Teilcircuit unter v zu 1 auswertet. Seien also $N = \{F_v, T_v \mid v \in V\}$ und $S = T_{v_o}$. Die Menge der Produktionen P geben wir für jeden Knoten einzeln an. Falls v ein Blattknoten ist, dann sind $F_v \to 0 \in P$ und $T_v \to 1 \in P$. Sei nun v ein innerer Knoten, wobei wir die eingehenden Knoten mit v' und v'' bezeichnen. Dann erhalten wir folgende Produktionen:

- Falls $s(v) = \wedge$, dann sind
  - $F_v \to F_{v'} F_{v''} \in P$,
  - $F_v \to F_{v'} T_{v''} \in P$,
  - $F_v \to T_{v'} F_{v''} \in P$,
  - $T_v \to T_{v'} T_{v''} \in P$.
- Falls $s(v) = \vee$, dann sind
  - $F_v \to F_{v'} F_{v''} \in P$,
  - $T_v \to F_{v'} T_{v''} \in P$,
  - $T_v \to T_{v'} F_{v''} \in P$,
  - $T_v \to T_{v'} T_{v''} \in P$.

Die Grammatik G und das Blattwort w lassen sich beide in logarithmischem Platz konstruieren (während man über den Circuit in Preorder iteriert). Damit haben wir gezeigt, dass das Wortproblem für KFGs **P**-schwer ist. Dass es überhaupt in **P** liegt, wird zum Beispiel durch den CYK-Algorithmus aus GTI gezeigt.