Einführung in Berechenbarkeit, Komplexität und Formale Sprachen: Blatt 10

Yascha Cebeci (6254677)
Bernhard Dietrich (6256800)
Lars Fernhomberg (6256030)
Sebastian Kniesburges (6257120)

Übungsgruppe 4
Montag 16:00-18:00
Gunnar Schomaker

Aufgabe 37 (5 Punkte)
In der Vorlesung wurden die beiden Probleme $CLIQUE$ und IndependentSet (IS) vorgestellt.
Des weiteren wurde bereits gezeigt, dass $CLIQUE$ NP-vollständig ist.
Zeigen Sie nun: $CLIQUE \leq_p IS$
Zu Zeigen: $CLIQUE \leq_p IS$

⇨ Es muss eine berechenbare Funktion Φ geben für die gilt:

$\langle G \rangle, k \in CLIQUE \rightarrow \Phi(\langle G \rangle k) \in IS$
$\langle G \rangle, k \notin CLIQUE \rightarrow \Phi(\langle G \rangle k) \notin IS$

Lösung:
- Wenn wir einen Graphen $G = (V, E)$ mit einer Zahl $k \in \{0, ..., |V|\}$ so dass gilt: $\langle G \rangle k \in CLIQUE$ dann brauch die Funktion Φ nichts anders zu tun als den Graphen zu invertieren.
- D.h.: Jede Kannte $(v_i, v_j) \in E$ für $i \neq j$ und $v_i, v_j \in V$ muss gelöscht werden und zwischen allen Knoten für die gilt $v_x, v_y \in V$ und $(v_x, v_y) \notin E$ muss erzeugt werden.

"$\Rightarrow$"
Der Graph besaß eine $CLIQUE$ der Grösse k. D.h.: es gab k Knoten bei denen jeder mit jedem Verbunden war. Jetzt wurde der Graph durch Φ wie geschildert invertiert und alle diese Kanten wurden gelöscht
$\Rightarrow$ Es gibt ein IS der Grösse k, Bei dem genau die k Knoten NICHT miteinander verbunden sind. $\langle G \rangle, k \in CLIQUE \rightarrow \Phi(\langle G \rangle k) \in IS$

"$\Leftarrow$"
Der Graph besaß keine $QLIQUE$ der Grösse k. D.h.: nach seiner Invertierung kann es ebenso wenig ein IS der Grösse k geben, bei dem genau k Knoten NICHT miteinander verbunden sind.
$\Rightarrow$ $\langle G \rangle, k \notin CLIQUE \rightarrow \Phi(\langle G \rangle k) \notin IS$

$\Rightarrow$ Das $IndependentSet$ Problem ist *NP(vollständig).*
Außerdem ist die Funktion Φ offensichtlich total berechenbar.

Das Problem $CLIQUE$ ist gegeben durch

$CLIQUE := \{(C(G), k) | G \text{ ist ein Graph}, k \in \mathbb{N}, G \text{ enthält eine } k-Clique\}$.

Das Problem Independent Set (IS) ist gegeben durch
$IS := \left\{\langle G\rangle, k \middle| G \text{ enthält eine unabhängige Menge der Größe } k\right\}$.

Damit nun das Problem IS auf das Problem $CLIQUE$ reduzierbar ist, muss es eine berechenbare Funktion $f(x)$ geben, für die mit $w \in IS \quad f(w) \in CLIQUE$ gilt.

Aufgabe 38 (5 Punkte)

$k-SAT$ ist das Erfüllbarkeitsproblem unter der Einschränkung, dass alle Klauseln genau k Literale enthalten. Also ist z.B. $(x_1 \vee \overline{x_2} \vee x_4) \wedge (x_1 \vee \overline{x_5} \vee x_6) \in 3-SAT$

a) Zeigen Sie, dass für $k > 3$ gilt:

$\bigvee_{i=1}^{k} z_i$ ist erfüllbar $\Leftrightarrow$ $\exists y_1, ..., y_{k-3}$:

$(z_1 \vee z_2 \vee y_1) \wedge \left(\bigwedge_{i=1}^{k-4} (\overline{y_i} \vee z_{i+2} \vee y_{i+1})\right) \wedge (\overline{y_{k-3}} \vee z_{k-1} \vee z_k)$ ist erfüllbar

z.Z: $Z := \bigvee_{i:=1}^{k} z_i$ ist erfüllbar

$\Leftrightarrow$ $\exists y_1, ..., y_{k-3} : (z_1 \vee z_2 \vee y_1) \wedge (\bigwedge_{i:=1}^{k-4} (\overline{y}_i, z_{i+2}, y_{i+1})) \wedge (\overline{y}_{k-3}, z_{k-1}, z_k)$ ist erfüllbar

"$\Rightarrow$"

Mindestens ein z_i ist $\neq 0$.

$\Rightarrow$ mindestens in der Klausel in der dieses wahre z_i steht kann $\overline{y}_i = 0$ und $y_{i+1} = 0$ sein.

$\Rightarrow$ wie eine Kettenreaktion ergibt sich daraus aus den weiteren Klauseln das dann das $\overline{y}_{i+1}$ der nachfolgenden Klausel und das y_i der davorstehenden Klausel wahr sind, usw.

$\Rightarrow$ alle weiteren Klauseln werden unabhängig von den restlichen z_i wahr.

$\Rightarrow$ $\exists y_1, ..., y_{k-3} : (z_1 \vee z_2 \vee y_1) \wedge (\bigwedge_{i:=1}^{k-4} (\overline{y}_i, z_{i+2}, y_{i+1})) \wedge (\overline{y}_{k-3}, z_{k-1}, z_k)$ ist erfüllbar.

"$\Leftarrow$"

Wenn $(z_1 \vee z_2 \vee y_1) \wedge \left(\bigwedge_{i=1}^{k-4} (\overline{y_i} \vee z_{i+2} \vee y_{i+1})\right) \wedge (\overline{y_{k-3}} \vee z_{k-1} \vee z_k)$ erfüllbar ist, so bedeutet dies, dass jede einzelne Klausel erfüllbar sein muss, da ansonsten die Konjunktion den gesamten Ausdruck falsifiziert.

Die Erfüllbarkeit kann nicht nur von y_i abhängen, da jedes y_i in der nächsten Klausel verneint wird. Da in der ersten und in der letzten Klausel jeweils nur ein y_i vorkommt, ist dieses die Negation des entsprechenden Wertes von y_i in der nächsten bzw. in der vorherigen Klausel. Dieses bedeutet nun, dass, wenn $y_1 = ... = y_{k-3} = 1$, mindestens eines der beiden Literale z_{k-1} bzw. z_k den Wert 1 annehmen müssen. Wenn $y_1 = ... = y_{k-3} = 0$, so bedeutet dies, dass mindestens eines der beiden Literale z_1 bzw. z_2 den Wert 1 annehmen müssen. Eine andere Permutation der Werte von y_i erzwingt automatisch, dass $y_i = 1$ und $y_{i+1} = 0$. Nun entsteht hierdurch eine Klausel $(0 \vee z_{i+2} \vee 0)$, so dass die Erfüllbarkeit der Klausel von z_{i+2} abhängt.

Wir haben nun gezeigt, dass, damit $(z_1 \vee z_2 \vee y_1) \wedge \left(\bigwedge_{\hat{i}=1}^{k-4} (\overline{y_i} \vee z_{i+2} \vee y_{i+1}) \right) \wedge (\overline{y_{k-3}} \vee z_{k-1} \vee z_k)$ erfüllbar ist, mindestens ein z_i erfüllbar ist. Da bei einer Disjunktion von verschiedenen Literalen bereits ein erfüllbares Literal genügt um die gesamte Disjunktion zu erfüllen ist somit automatisch $\bigvee_{i=1}^{k} z_i$ erfüllt, so dass die Aussage wahr ist.

b) Denken Sie sich zu Klauseln der Länge eins und zwei Klauseln der Länge drei aus, die immer genau dann erfüllbar sind, wenn die kurzen Klauseln erfüllbar sind.

Gemäß Aussage in der Übung ist hier lediglich ein Beispiel anzugeben, welches die angegebene Bedingung erfüllt.

Klausel $K_1 = x_1$
Klausel $K_2 = x_2 \vee x_3$
Daraus resultiert Klausel $K_3 = x_1 \vee x_2 \vee x_3$, welche erfüllbar ist, wenn die Klausel K_1 oder die Klausel K_2 erfüllbar ist.

Aufgabe 38 (5 Punkte)
Zeigen Sie, dass $2-SAT \in P$ gilt.

Klausen, bei denen ein Literal zweimal auftaucht bewirken, dass dieses Literal den Wert 1 hat, so dass hierdurch auch andere Klauseln verkürzt werden. Dieser Vereinfachungsschritt wird nun solange ausgeführt, bis nur noch Klauseln vorliegen, die zwei verschiedene Literale haben. Das vorliegende Problem lässt sich nun durch einen gerichteten Graphen verdeutlichen, bei dem jede Klausel $a \vee b$ durch die beiden Aussagen $\overline{a} = 1 \Rightarrow b = 1$ und $\overline{b} = 1 \Rightarrow a = 1$ ersetzt werden kann, wobei die beiden Aussagen durch die beiden gerichteten Kanten $(\overline{a}, b)$ und $(\overline{b}, a)$ dargestellt werden. Wir nennen $(\overline{a}, b)$ dual zu $(\overline{b}, a)$ und umgekehrt.
In dem nun entstandenen Graphen lässt sich durch einen geeigneten Algorithmus (siehe Vorlesung DuA) in linearer Zeit eine starke Zusammenhangskomponente und einen dazu gehörigen Wald W finden.
Wenn x_i und $\overline{x_i}$ in einer starken Zusammenhangskomponente liegen, lassen sich beide Klauseln nicht gemeinsam erfüllen, da entweder $x_i = 1$ oder $\overline{x_i} = 1$ sein muss, woraus allerdings folgt, dass $\overline{x_i} = 1$ bzw. $x_i = 1$ sein muss, was in beiden Fällen ein offensichtlicher Widerspruch ist. Ansonsten gibt es eine erfüllende Belegung, die sich ermitteln lässt, indem ein Blatt B aus W gewählt wird und alle Literale der zugehörigen Komponente auf 1 gesetzt werden. In der dualen Komponente B^* müssen nun alle Literale auf 0 gesetzt werden. Die duale Komponente ist in W eine Wurzel. Aus W können die zwei Knoten B und B^* entfernt werden. Nach diesem Schritt wird entsprechend fortgefahren, bis am Schluss eine Belegung erschaffen wurde, die alle Aussagen, welche durch die Kanten ausgedrückt werden, erfüllt. Hierdurch sind dann automatisch auch alle andere Klauseln erfüllt.

Aufgabe 40 (5 Punkte)
In Satz 4.11 des Skriptes ist die Definition für das Problem $\mathrm{BinProgr}$ angegeben.
Zeigen Sie: $CLIQUE \leq_p \mathrm{BinProgr}$

Das Problem $\mathrm{BinProgr}$ ist gegeben durch

$\mathrm{BinProgr}{:=}\left\{(\mathrm{A,b}) \middle| A \in M_{n,m}(\mathbb{Z}), b \in \mathbb{Z}^n, \exists y \in \{0,1\}^n \text{ mit } A \cdot y \leq b\right\}$.

Das Problem $CLIQUE$ ist gegeben durch

$CLIQUE := \left\{(C(G),k) \middle| G \text{ ist ein Graph}, k \in \mathbb{N}, G \text{ enthält eine } k-Clique\right\}$.

Zu zeigen ist, dass das Problem $CLIQUE$ in polynomieller Zeit durch eine berechenbare Funktion auf das Problem $\mathrm{BinProg}$ reduzierbar ist.

In der Vorlesung wurde gezeigt, dass $SAT \leq_p \mathrm{BinProg}$. Da SAT, wie ebenfalls in der Vorlesung gezeigt, NP-vollständig ist und $CLIQUE \in NP$, muss auch die Reduktion $CLIQUE \leq_p SAT$ möglich sein.
Somit folgt $CLIQUE \leq_p SAT \leq_p \mathrm{BinProg}$, wodurch aufgrund der Transitivität der Reduktion $CLIQUE \leq_p \mathrm{BinProg}$ folgt.