Beweisskizze Lemma 9.13

Sei $\mathcal{L} \in NL$ und $M_{\mathcal{L}}$ eine NTM, die $\mathcal{L}$ mit log-space entscheidet.

Zeige folgende Punkte:

1. Zu $M_{\mathcal{L}}$ existiert $M'_{\mathcal{L}}$, sodass $M'_{\mathcal{L}}$ wie $M_{\mathcal{L}}$ arbeitet, am Ende aber an das linke Ende des Bandes zurückkehrt, und Arbeitsband löscht/leert.
2. Der Konfigurationsgraph G_M zu $M'_{\mathcal{L}}$ log-space berechenbar ist
3. $M'_{\mathcal{L}}$ akzeptiert x, gdw in G_M ein Pfad von s nach t existiert. Dabei entsprechen s und t der Start- bzw. Zielkonfiguration. $\square$

Kapitel 9.4 - 2SAT

2SAT

Seien $x_0, x_1, \dots, x_n$ eine Menge von boolschen Variablen

Ein Literal L ist von der Form x_i (positiv) oder $\neg x_i$ (negativ).

Eine Klausel ist eine Disjunktion von Literalen

$$C = L_1 \vee L_2 \vee L_3 \vee \dots \vee L_k$$

Eine aussagenlogische Formel in konjunktiver Normalform (CNF) ist eine Konjunktion von Klauseln

$$F = C_1 \wedge C_2 \wedge C_3 \wedge \dots C_m$$

Problem SAT

Gegeben: Formel F in CNF

Frage: Ist F erfüllbar, d.h. gibt es eine Belegung φ der Variablen, sodass $\varphi(F) = \text{true}$?

Problem k-SAT

Gegeben: Formel F in kCNF, d.h. jede Klausel hat $\leq k$ Literale.

Frage: Ist F erfüllbar?

1SAT ist trivial lösbar ($\exists L_1, L_2$ mit $L_1 = \neg L_2$?)

Wir betrachten nun 2SAT und zeigen, NL-Vollständigkeit

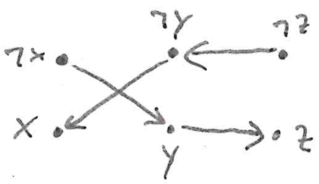
~~"Hardness"~~

Zunächst: Jede Formel F kann als Graph interpretiert werden. Klauseln der Form $(\neg x \vee y)$ sind äquivalent zu $x \Rightarrow y$ oder auch $\neg y \Rightarrow \neg x$. Das sind quasi Kanten!

$(x \vee y) \wedge (\neg y \vee z)$

$\equiv$

$\neg x \to y$
$\neg y \to x$
$y \to z$
$\neg z \to \neg y$



Kette von Implikationen bilden Pfade. $\neg z \Rightarrow \neg y \Rightarrow x$

Ist $\neg z = 1$ muss auch $\neg y = 1$ und $x = 1$!

Also; Konstruiere Graph G_F mit

- 2 Knoten v_t, v_f für ~~eine~~ jede Variable
- 2 Knoten $u \to v$ sind verbunden, wenn es eine Klausel der form $(\neg u \vee v)$ gibt.

→ Kann mit log-space berechnet werden!

Ist $L_1 \to L_2$, dann gilt auch $\neg L_2 \to \neg L_1$.

Oder mit Transitivität $L_1 \to^* L_2 \iff \neg L_2 \to^* \neg L_1$
wobei L_1 und L_2 unterschiedliche Variablen beinhalten (also $L_1 \neq \neg L_2$)

Zu Zeigen:

$$F \text{ unerfüllbar} \iff \exists x: x \to^* \neg x \text{ und } \neg x \to^* x$$

Skizze:

"$\Leftarrow$" Selbst (per ~~Wi Kontraposi~~ Widerspruch)

"$\Rightarrow$" Betrachte folgenden Algorithmus unter der Annahme, dass nicht beide Pfade existieren

1. while $\exists$ Literal ohne Wahrheitswert
2. wähle solches Literal L, sodass kein Pfad von L nach $\neg L$ existiert.
3. for Literal L' mit $L \to^* L'$ do
   Setze $\varphi(L') = 1$
4. for Literal L' mit $L' \to^* \neg L$ do
   Setze $\varphi(L') = 0$

Zu zeigen: die erhaltene Belegung ist gültig.

1. Jedes Literal erhält einen Wert.
2. Belegung ist wohldefiniert, d.h. es existiert kein Literal L' mit $\varphi(L') = \varphi(\neg L')$.
3. Alle Klauseln sind erfüllt.

Das überprüfen aller drei Punkte ist zur Übung ausgelassen. $\square$

Folgender Algorithmus löst nun $\overline{2SAT}$

1. Konstruiere G_F
2. for Variable x in F do
3. if $G_F \# x \# \neg x \in PATH$ und $G_F \# \neg x \# x \in PATH$ then
   return true
4. return false

$\Longrightarrow \overline{2SAT} \in NL \Longrightarrow 2SAT \in NL$ (Membership) (Lemma 9.19)

Lemma 9.23: 2SAT ist coNL-schwer

Beweisskizze:
Reduktion von $\overline{ACYCLICPATH}$ auf 2SAT
Setze Variablen $Vars = \{x \mid x \text{ ist Knoten von } G\}$
Für jede Kante $x \to y$ in G erstellen wir die Klausel $(\neg x \vee y)$
für s und t fügen wir die Klauseln (s) und $(\neg t)$ ein.
Zu zeigen (selbst)
1. F erfüllbar $\Longleftrightarrow$ G besitzt keinen s-t-Pfad.
2. Reduktion ist log-space berechenbar