

Theorie der Informatik

IV.4. Satz von Cook und Levin

Malte Helmert Christian Tschudin

Universität Basel

13. Mai 2013

Theorie der Informatik

13. Mai 2013 — IV.4. Satz von Cook und Levin

IV.4.1 Satz von Cook und Levin

Überblick: Vorlesung

Vorlesungsteile

- I. Logik
- II. Automatentheorie und formale Sprachen
- III. Berechenbarkeitstheorie
- IV. **Komplexitätstheorie** ← Sie sind hier!

Überblick Komplexitätstheorie

Überblick über diesen Vorlesungsteil:

- IV. Komplexitätstheorie
 - IV.1. Motivation und Einführung
 - IV.2. Nichtdeterminismus
 - IV.3. P, NP und polynomielle Reduktionen
 - IV.4. **Satz von Cook und Levin** ← hier!
 - IV.5. einige NP-vollständige Probleme

IV.4.1 Satz von Cook und Levin

SAT ist NP-vollständig

Definition (SAT)

Das Problem **SAT** (Erfüllbarkeit = satisfiability) ist wie folgt definiert:

Gegeben: eine aussagenlogische Formel φ

Frage: Ist φ erfüllbar?

Satz (Cook, 1971; Levin, 1973)

SAT ist NP-vollständig.

Beweis.

SAT $\in$ NP: Raten und prüfen.

SAT ist NP-hart: etwas komplizierter (wird fortgesetzt) ...

NP-Härte von SAT (1)

Beweis (Fortsetzung).

Wir müssen zeigen: $A \leq_p \text{SAT}$ für alle $A \in \text{NP}$.

Sei A ein beliebiges Problem in NP.

Wir müssen eine polynomielle Reduktion von A auf SAT finden, also eine in polynomieller Zeit berechenbare Funktion f , so dass für jedes Eingabewort w für A gilt:

$w \in A$ gdw. $f(w)$ ist eine erfüllbare aussagenlogische Formel. ...

NP-Härte von SAT (2)

Beweis (Fortsetzung).

Wegen $A \in \text{NP}$ gibt es eine NTM M und ein Polynom p , so dass M das Problem A in Zeit p akzeptiert.

Idee: Konstruiere eine Formel, die **die möglichen Konfigurationen** kodiert, die M auf Eingabe w in Zeit $p(|w|)$ durchlaufen kann, und die **genau dann erfüllbar** ist, wenn in dieser Zeit **eine Endkonfiguration erreicht werden kann**. ...

NP-Härte von SAT (3)

Beweis (Fortsetzung).

Sei $M = \langle Z, \Sigma, \Gamma, \delta, z_0, \square, E \rangle$ eine NTM für A
und sei p ein Polynom, das die Laufzeit von M beschränkt.

Sei $w = w_1 \dots w_n \in \Sigma^*$ die Eingabe für M .

Nummerieren wir die Bandpositionen mit ganzen Zahlen (positiv und negativ), so dass der TM-Kopf zu Beginn auf Position 1 steht.

Beobachtung: innert $p(n)$ vielen Berechnungsschritten kann der TM-Kopf nur Positionen aus der Menge
 $Pos = \{-p(n) + 1, -p(n) + 2, \dots, -1, 0, 1, \dots, p(n) + 1\}$ erreichen.

Statt unendlich vieler Bandpositionen müssen wir nur diese (polynomiell vielen!) Positionen berücksichtigen. ...

NP-Härte von SAT (4)

Beweis (Fortsetzung).

Wir können Konfigurationen von M kodieren, indem wir angeben:

- ▶ was der aktuelle **Zustand** von M ist
- ▶ auf welche Zelle in Pos der **TM-Kopf** zeigt
- ▶ welche **Symbole** aus Γ das **Band** auf den Positionen in Pos beinhaltet

$\rightsquigarrow$ durch Aussagevariablen kodierbar

Um nicht nur eine Konfiguration, sondern eine **Berechnung** zu kodieren, benötigen wir **eine Kopie** dieser Variablen für jeden Schritt der Berechnung.

Wir müssen dabei nur die Berechnungsschritte
 $Steps = \{0, 1, \dots, p(n)\}$ berücksichtigen,
denn M soll innert $p(n)$ Schritten akzeptieren. ...

NP-Härte von SAT (5)

Beweis (Fortsetzung).

Verwende die folgenden Aussagevariablen in der Formel $f(w)$:

- ▶ $state_{t,z}$ ($t \in Steps, z \in Z$)
 $\rightsquigarrow$ kodiert Zustand der NTM in der t -ten Konfiguration
- ▶ $head_{t,i}$ ($t \in Steps, i \in Pos$)
 $\rightsquigarrow$ kodiert Kopfposition in der t -ten Konfiguration
- ▶ $tape_{t,i,a}$ ($t \in Steps, i \in Pos, a \in \Gamma$)
 $\rightsquigarrow$ kodiert Bandinhalt in der t -ten Konfiguration

Konstruiere $f(w)$ so, dass jede erfüllende Belegung

- ▶ eine **Folge von Konfigurationen** der TM beschreibt,
 - ▶ die **mit der Startkonfiguration beginnt**,
 - ▶ eine **akzeptierende Konfiguration erreicht**
 - ▶ und **den TM-Regeln in δ folgt**
- ...

NP-Härte von SAT (6)

Beweis (Fortsetzung).

Kleiner Trick: Modifiziere M so, dass sie in jedem Zustand die Möglichkeit hat, **nichts zu tun**.

- ▶ Füge δ Transitionen der Form $\langle \langle z, a \rangle, \langle z, a, N \rangle \rangle$ für jedes z und a hinzu.
 - ▶ Ändert das Akzeptanz-Verhalten von M nicht und erlaubt uns anzunehmen, dass eine akzeptierende Berechnung **genau** $p(n)$ Schritte braucht statt **höchstens** $p(n)$ Schritte.
 - ▶ Das macht uns im Folgenden das Leben etwas leichter.
- ...

NP-Härte von SAT (7)

Beweis (Fortsetzung).

Hilfsformel:

$$\text{oneof } X := \left(\bigvee_{x \in X} x \right) \wedge \neg \left(\bigvee_{x \in X} \bigvee_{y \in X \setminus \{x\}} (x \wedge y) \right)$$

1. Beschreibe Folge von Konfigurationen der TM:

$$\text{Valid} := \bigwedge_{t \in \text{Steps}} \left(\text{oneof} \{ \text{state}_{t,z} \mid z \in Z \} \wedge \right. \\ \left. \text{oneof} \{ \text{head}_{t,i} \mid i \in \text{Pos} \} \wedge \right. \\ \left. \bigwedge_{i \in \text{Pos}} \text{oneof} \{ \text{tape}_{t,i,a} \mid a \in \Gamma \} \right) \dots$$

NP-Härte von SAT (8)

Beweis (Fortsetzung).

2. Beginne in Startkonfiguration

$$\text{Init} := \text{state}_{0,z_0} \wedge \text{head}_{0,1} \wedge \bigwedge_{i=1}^n \text{tape}_{0,i,w_i} \wedge \bigwedge_{i \in \text{Pos} \setminus \{1, \dots, n\}} \text{tape}_{0,i,\square} \dots$$

NP-Härte von SAT (9)

Beweis (Fortsetzung).

3. Erreiche eine akzeptierende Konfiguration

$$\text{Accept} := \bigvee_{q_e \in E} \text{state}_{p(n),q_e} \dots$$

NP-Härte von SAT (10)

Beweis (Fortsetzung).

4. Folge den Regeln in δ :

$$\text{Trans} := \bigwedge_{t \in \text{Steps}} \left(\bigvee_{q_e \in E} \text{state}_{t,q_e} \vee \bigvee_{R \in \delta} \text{Rule}_{t,R} \right)$$

wobei...

...

NP-Härte von SAT (11)

Beweis (Fortsetzung).

4. Folge den Regeln in δ (Fortsetzung):

$$\begin{aligned} \text{Rule}_{t, \langle \langle z, a \rangle, \langle z', a', y \rangle \rangle} := \\ \text{state}_{t,z} \wedge \text{state}_{t+1,z'} \wedge \\ \bigwedge_{i \in \text{Pos}} (\text{head}_{t,i} \rightarrow \text{tape}_{t,i,a} \wedge \text{head}_{t+1,i+y} \wedge \text{tape}_{t+1,i,a'}) \wedge \\ \bigwedge_{i \in \text{Pos}} \bigwedge_{a'' \in \Gamma} (\neg \text{head}_{t,i} \wedge \text{tape}_{t,i,a''} \rightarrow \text{tape}_{t+1,i,a''}) \end{aligned}$$

(Spezialfall: *tape*- und *head*-Variablen, bei denen der Bandindex $i + y$ ausserhalb von *Pos* liegt, durch $\perp$ ersetzen; ebenso Variablen, bei denen der Zeitindex ausserhalb von *Steps* liegt.) ...

NP-Härte von SAT (12)

Beweis (Fortsetzung).

Alles zusammengesetzt:

Setze $f(w) := \text{Valid} \wedge \text{Init} \wedge \text{Accept} \wedge \text{Trans}$.

- ▶ $f(w)$ kann in polynomieller Zeit in $|w|$ konstruiert werden.
- ▶ $w \in A$ gdw. M akzeptiert w in $p(|w|)$ Schritten
 gdw. $f(w)$ ist erfüllbar
 gdw. $f(w) \in \text{SAT}$

$\rightsquigarrow A \leq_p \text{SAT}$

Da $A \in \text{NP}$ beliebig war, gilt dies für jedes $A \in \text{NP}$.

Folglich ist SAT NP-hart und damit auch NP-vollständig. □