

Theorie der Informatik

21. Satz von Cook und Levin

Malte Helmert Gabriele Röger

Universität Basel

18. Mai 2015

Theorie der Informatik

18. Mai 2015 — 21. Satz von Cook und Levin

21.1 Satz von Cook und Levin

21.2 Zusammenfassung

Überblick: Vorlesung

Vorlesungsteile

- I. Logik ✓
- II. Automatentheorie und formale Sprachen ✓
- III. Berechenbarkeitstheorie ✓
- IV. **Komplexitätstheorie**

Überblick: Komplexitätstheorie

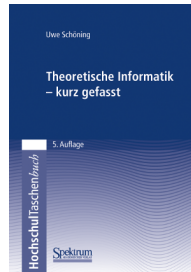
IV. Komplexitätstheorie

- 19. Motivation und Einführung ✓
- 20. P, NP und polynomielle Reduktionen ✓
- 21. **Satz von Cook und Levin**
- 22. einige NP-vollständige Probleme

Literatur zu diesem Vorlesungskapitel

Theoretische Informatik - kurz gefasst
von Uwe Schöning (5. Auflage)

► Kapitel 3.2



21.1 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 $\text{Pos} = \{-p(n)+1, -p(n)+2, \dots, -1, 0, 1, \dots, p(n)+1\}$ 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 $\text{Steps} = \{0, 1, \dots, p(n)\}$ berücksichtigen, denn M soll innert $p(n)$ Schritten akzeptieren. . . .

NP-Härte von SAT (5)

Beweis (Fortsetzung).

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

- ▶ $\text{state}_{t,z}$ ($t \in \text{Steps}$, $z \in Z$)
 $\rightsquigarrow$ kodiert Zustand der NTM in der t -ten Konfiguration
- ▶ $\text{head}_{t,i}$ ($t \in \text{Steps}$, $i \in \text{Pos}$)
 $\rightsquigarrow$ kodiert Kopfposition in der t -ten Konfiguration
- ▶ $\text{tape}_{t,i,a}$ ($t \in \text{Steps}$, $i \in \text{Pos}$, $a \in \Gamma$)
 $\rightsquigarrow$ 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:

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

...

NP-Härte von SAT (8)

Beweis (Fortsetzung).

2. Beginne in Startkonfiguration

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

...

NP-Härte von SAT (9)

Beweis (Fortsetzung).

3. Erreiche eine akzeptierende Konfiguration

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

...

NP-Härte von SAT (10)

Beweis (Fortsetzung).

4. Folge den Regeln in δ :

$$Trans := \bigwedge_{t \in Steps} \left(\bigvee_{q_e \in E} state_{t,q_e} \vee \bigvee_{R \in \delta} Rule_{t,R} \right)$$

wobei...

...

NP-Härte von SAT (11)

Beweis (Fortsetzung).

4. Folge den Regeln in δ (Fortsetzung):

$$\begin{aligned} Rule_{t, \langle \langle z, a \rangle, \langle z', a', y \rangle \rangle} := & \\ & state_{t,z} \wedge state_{t+1,z'} \wedge \\ & \bigwedge_{i \in Pos} (head_{t,i} \rightarrow tape_{t,i,a} \wedge head_{t+1,i+y} \wedge tape_{t+1,i,a'}) \wedge \\ & \bigwedge_{i \in Pos} \bigwedge_{a'' \in \Gamma} (\neg head_{t,i} \wedge tape_{t,i,a''} \rightarrow tape_{t+1,i,a'') \end{aligned}$$

(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 $f(w) := Valid \wedge Init \wedge Accept \wedge Trans$.

- ▶ $f(w)$ 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. $f(w) \in SAT$

$\rightsquigarrow A \leq_p SAT$

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

Folglich ist SAT NP-hart und damit auch NP-vollständig. $\square$

21.2 Zusammenfassung

Zusammenfassung

- ▶ Das Erfüllbarkeitsproblem der Aussagenlogik (**SAT**) ist NP-vollständig.
- ▶ **Beweisidee** für die **NP-Härte**:
 - ▶ Jedes Problem in NP kann durch eine NTM in polynomieller Zeit $p(|w|)$ bei Eingabe w gelöst werden.
 - ▶ Stelle zu Wort w eine aussagenlogische Formel φ auf, die die Berechnungsschritte der NTM auf Eingabe w kodiert.
 - ▶ Konstruiere φ so, dass es genau dann erfüllt werden kann, wenn es eine akzeptierende Berechnung der Länge $p(|w|)$ gibt.