

Komplexitätstheorie

WiSe 2009/10 in Trier

Henning Fernau
Universität Trier
fernau@uni-trier.de

Komplexitätstheorie Gesamtübersicht

- Organisatorisches / Einführung
Motivation / Erinnerung / Fragestellungen
- Diskussion verschiedener Komplexitätsklassen:
Zeitkomplexität
Platzkomplexität
- zugehörige Reduktionsbegriffe
- vollständige Probleme
- Anpassung von Klassenbegriffen und Reduktionen

Zwei konkrete (many-one) Reduktionen

Gegeben $A \subseteq W(\Sigma)$, $B \subseteq W(\Delta)$

- A heißt *Polynomzeit-reduzierbar* auf B (in Zeichen $A \leq_p B$), wenn für ein $k \in \mathbb{N}$ eine Funktion $f(n) \in \text{FTIME}(n^k)$ mit

$$\forall x \in W(\Sigma) (x \in A \Leftrightarrow f(x) \in B)$$

existiert.

- A heißt *mit logarithmischem Platz reduzierbar* oder *logspace-reduzierbar* auf B (in Zeichen $A \leq_{\log} B$), wenn eine Funktion $f \in \text{FSPACE}(\log n)$ mit

$$\forall x \in W(\Sigma) (x \in A \Leftrightarrow f(x) \in B)$$

existiert.

Lemma 1

\leq_p und \leq_{\log} sind reflexiv und transitiv, d.h. *Vorordnungen*; solche Relationen werden auch *Quasiordnungen* oder *Präordnungen* genannt.

Beweis: Die Identität ist (sogar) mit konstantem Platz und linearer Zeit berechenbar, was sofort die Reflexivität liefert.

Betrachte $A \subseteq \Sigma^*$, $B \subseteq \Gamma^*$, $C \subseteq \Delta^*$. Seien M, N det. TM mit $f_M : \Sigma^* \rightarrow \Gamma^*$ und $f_N : \Gamma^* \rightarrow \Delta^*$.

Gilt $x \in A \iff f_M(x) \in B$ sowie $y \in B \iff f_N(y) \in C$, so gilt für die Komposition g von f_M und f_N : $x \in A \iff g(x) = f_N(f_M(x)) \in C$.

Betrachte \hat{M} : \hat{M} simuliert zunächst M auf der Eingabe x , schreibt deren Ausgabe aber auf ein spezielles Arbeitsband. Danach simuliert \hat{M} die TM N auf der Eingabe $f_M(x)$.

Offenbar gilt: $g = f_{\hat{M}}$.

$T_{\hat{M}}(x) \leq cT_M(x) + cT_N(f_M(x))$. Da $\lg(f_M(x)) \leq T_M(x)$, folgt, dass \hat{M} Polynomzeit benötigt, falls M und N dies tun.

$S_{\hat{M}}(x) \leq \hat{c}S_M(x) + S_N(f_M(x)) + \lg(f_M(x))$.

Problem: $\lg(f_M(x)) > S_M(x)$ möglich, z.B. für $f_M = \text{id}_{\Sigma}$!

Logspace-Tricks

Grundidee: Vermeide Speichern des Ergebnisses $f_M(x)$, indem ggf. immer wieder **Neuberechnungen** angestoßen werden.

Konkret: Will man das p -te Bit der Ausgabe von $T_M(x)$ lesen, so simuliert die konstruierte det. TM M' M solange, bis das p -te Bit ausgegeben würde. Die Ausgabe von M wird dabei nicht explizit aufgeschrieben, es wird lediglich (mit Hilfe eines Zählers p') mitprotokolliert, um zu entdecken, wann M das p -te Zeichen auf das Ausgabeband schreibe.

\leadsto **Zwei Zähler p, p'** sind nötig, die bis max. $\lg(f_M(x))$ zählen können müssen.

Damit gilt: $S_{M'}(x) \leq c'S_M(x) + S_N(f_M(x)) + 2 \log(\lg(f_M(x)))$.

Wegen $\lg(f_M(x)) \leq T_M(x) \leq d^{S_M(x)}$ (Zusammenraum Zeit und Platz) folgt:

$\log(\lg(f_M(x))) \leq dS_M(x)$. $\leadsto S_N(f_M(x)) \leq c_N \log(\lg(f_M(x))) \leq c_N \cdot d \cdot S_M(x)$

Aus $S_M(x) \leq c_S \log(\lg(x))$ folgt somit:

$S_{M'}(x) \leq c'S_M(x) + c_N \cdot d \cdot S_M(x) + dS_M(x) \leq c'_S \log(\lg(x));$ q.e.d.

Beispiel: $A = \{x \in W(\Sigma) \mid x = x^r\}$, $B = \{xx \mid x \in W(\Sigma)\}$ wie oben.

Definiere $g: W(\Sigma) \rightarrow W(\Sigma)$ durch

$$g(xy) := xy^r$$

für Worte $w = xy$ mit gerader Länge, d.h. $\lg(x) = \lg(y)$, bzw.

$$g(w) := 10$$

für Worte w mit ungerader Länge $\lg(w)$

Dann gilt $g \in \text{FSPACE}(\log n)$ und

$$w \in B \Leftrightarrow (\exists x)w = xx \Leftrightarrow (\exists x)g(w) = xx^r \Leftrightarrow g(w) \in A$$

Also $B \leq_{\log} A$.

D.h. $A \leq_{\log} B$ und $B \leq_{\log} A$, obwohl $A \neq B$ ist.

$\Rightarrow \leq_{\log}$ ist nicht antisymmetrisch, d.h. keine (Halb-)Ordnungsrelation.

Zusammenhang der Reduktionsbegriffe

Erinnerung

Lemma 2 Sei M eine (nichtdeterministische) k -Band-Turingmaschine. Dann gibt es Konstanten c_1, c_2 , sodass für alle $x \in L_M$ gilt:

$$T_M(x) \leq c_1^{S_M(x)} \cdot \lg(x) + c_2$$

Mit Beweis von Lemma 2 folgt für Reduktionen:

Lemma 3

$$A \leq_{\log} B \implies A \leq_p B$$

Offene Frage: Umkehrung dieses Ergebnisses?

Lemma 4 Sei $\mathcal{K} \in \{\mathbf{P}, \mathbf{NP}, \mathbf{PSPACE}\}$. Dann gilt:

$$A \leq_p B \wedge B \in \mathcal{K} \implies A \in \mathcal{K}$$

Sei $\mathcal{K}' \in \{\mathbf{L}, \mathbf{NL}, \mathbf{P}, \mathbf{NP}, \mathbf{PSPACE}\}$. Dann gilt:

$$A \leq_{\log} B \wedge B \in \mathcal{K}' \implies A \in \mathcal{K}'$$

Beweisidee: Ähnlich wie Nachweis der Transitivität der Reduktionen.

\leadsto *Abschluss* der Komplexitätsklassen nach unten

i.d.R. Nachweis von $A \in \mathcal{K}$ über $A \leq_{\log} B$ bzw. $A \leq_p B$ für ein $B \in \mathcal{K}$.

Beispiel: `INDEPENDENT_SET(k)` (zu festem $k \in \mathbb{N}$):

- **Gegeben:** ungerichteter Graph $G = (V, E)$ (durch Adjazenzmatrix)
- **Frage:** Gibt es eine Menge von k Knoten in G , von denen keine zwei benachbart sind?

Reduktion auf `CLIQUE(k)`-Problem: Betrachte Funktion f mit: f invertiert alle 0en und 1en mit Ausnahme der Diagonalen \rightsquigarrow

— x Adj.-matrix eines unger. Graphen G g.d.w. $f(x)$ Adj.-matrix eines unger. Graphen G'

—Kante $\{z, z'\}$ existiert in G g.d.w. Kante $\{z, z'\}$ existiert nicht in G' .

\Rightarrow Knoten $\{z_1, \dots, z_k\}$ in G paarweise nicht benachbart g.d.w. sie bilden in G' eine k -Clique.

Mitgliedschaft in Logspace

$$x \in \text{INDEPENDENT_SET}(k) \iff f(x) \in \text{CLIQUE}(k)$$

Mit $f \in \text{FSPACE}(\log n)$:

$$\text{INDEPENDENT_SET}(k) \leq_{\log} \text{CLIQUE}(k)$$

Mit $\text{CLIQUE}(k) \in \mathbf{L}$ folgt aus vorigem Lemma:

$$\text{INDEPENDENT_SET}(k) \in \mathbf{L}$$

Komplexitätsklassen und Reduktionen

Mit Lemma 1 (also für geeignete Klassen und Reduktionen) gilt:

$$B \in \mathcal{K} \Rightarrow \{A \mid A \leq B\} \subseteq \mathcal{K}$$

für \mathcal{K} und \leq aus Lemma 4. Trivialerweise:

$$B \in \mathcal{K} \Leftarrow \{A \mid A \leq B\} \subseteq \mathcal{K}$$

$$\rightsquigarrow B \in \mathcal{K} \Leftrightarrow \{A \mid A \leq B\} \subseteq \mathcal{K}$$

Per Definition: B ist schwerstes (härtestes) Problem für $\{A \mid A \leq B\}$.

\rightsquigarrow **Frage:** Gibt es zu den genannten Komplexitätsklassen \mathcal{K} und Relationen \leq “schwerste Elemente” / härteste Probleme, und wenn ja, wie sehen sie aus?

Hart vs. vollständig

Sei \mathcal{K} eine Klasse von Problemen. Ein Problem B heißt

- \mathcal{K} -hart in Bezug auf Polynomzeit-Reduktionen, wenn $\mathcal{K} \subseteq \{A \mid A \leq_p B\}$ gilt.
 \mathcal{K} -hart in Bezug auf logspace-Reduktionen, wenn $\mathcal{K} \subseteq \{A \mid A \leq_{\log} B\}$ gilt.
- \mathcal{K} -vollständig in Bezug auf Polynomzeit-Reduktionen, wenn B \mathcal{K} -hart für Polynomzeit-Reduktionen ist und zudem $B \in \mathcal{K}$ gilt.
 \mathcal{K} -vollständig in Bezug auf logspace-Reduktionen, wenn B \mathcal{K} -hart für logspace-Reduktionen ist und zudem $B \in \mathcal{K}$ gilt.

Reduktionsrelation \leq_p bzw. \leq_{\log} oft aus Kontext ersichtlich!
Im Zweifelsfalle *stärkere* Relation \leq_{\log} gemeint!

Bsp.: B sei beliebiges **NP**-vollständiges Problem (für \leq_p) Dann gilt:

$$\mathbf{NP} \subseteq \{A \mid A \leq_p B\}$$

Andererseits folgt mit $B \in \mathbf{NP}$ aus Lemma 4 auch:

$$\{A \mid A \leq_p B\} \subseteq \mathbf{NP}$$

\rightsquigarrow Für jedes **NP**-vollständiges Problem B gilt:

$$\{A \mid A \leq_p B\} = \mathbf{NP},$$

\rightsquigarrow durch B und \leq_p ist die (nichtdet.!) Klasse **NP** eindeutig festgelegt.

Zudem: Falls $B \in \mathbf{P}$ für B beweisbar gilt, folgt $\mathbf{NP} = \mathbf{P}$.

Beispiel: $\mathbf{NL} \subseteq \mathbf{P}$ über **NL**-Vollständigkeit (bzgl. \leq_{\log}) von GAP.

Seien Σ und $L \subseteq W(\Sigma)$ gegeben. Dann ist die *charakteristische Funktion* $\chi_L : \Sigma^* \rightarrow \{0, 1\}$ von L definiert durch:

$$\chi_L(x) := \begin{cases} 1, & x \in L \\ 0, & x \notin L \end{cases}$$

Zunächst als einfacher Fall: **P-vollständige Mengen bzgl. \leq_p**

Lemma 5 *Seien Σ und $L \subseteq W(\Sigma)$, $L \in \mathbf{P}$, gegeben. Es sei $\Delta := \{0, 1\}$. Dann gibt es ein k mit $\chi_L \in \text{FTIME}(n^k)$ für die charakteristische Funktion von L , aufgefasst als $\chi_L : W(\Sigma) \rightarrow W(\Delta)$.*

Beweis von Lemma 5

$L \in \mathbf{P} \Rightarrow \exists \text{TM } M \exists c, k \forall x \in L : T_M(x) \leq (\lg(x))^k + c.$

Problem: Unklar, was bei $x \notin L$ passiert (Endlosschleife).

Lösung: Richte Binärzähler in simulierender TM M' ein.

M' schreibt zunächst Wort 10^m mit $m = k \log(\lg(x) + 1) + \log c$ auf ein Arbeitsband.

M' simuliert M ohne Beachtung der möglichen Ausgabe von M .

Bei jedem Simulationsschritt wird der Zähler dekrementiert.

Wird der Zähler auf Null heruntergezählt oder bricht die Berechnung von M vorher ohne Ergebnis ab, so gibt die simulierende Maschine eine Null aus und hält, andernfalls (d.h., M erreicht eine Endkonfiguration), wird Eins ausgegeben und gehalten.

Hält M' unter Ausgabe einer Eins auf Eingabe x , so benötigt sie $\mathcal{O}(T_M(x))$ Zeit für die eigentliche Simulation und (amortisiert, s. Analyse oben) $\mathcal{O}((\lg(x))^k + c)k \log(\lg(x) + 1) + \log c$ Zeit für die Verwaltung des Binärzählers. $\rightsquigarrow T_{M'}(x) \leq c_1(\lg(x))^k + c'_1$ für geeignetes c_1 .

Andernfalls, d.h., wenn $x \notin L$, werden schlimmstenfalls $(\lg(x) + 1)^k + c$ Simulationsschritte durchgeführt.

Zusammen mit Binärzählerverwaltung $\rightsquigarrow T_{M'}(x) \leq c_2(\lg(x))^k + c'_2$ für geeignetes c_2 .

Also gilt: $\chi_L = f_{M'} \in \text{FTIME}(n^k)$.

Mit analogem Beweis:

Folgerung 6 Seien Σ und $L \subseteq W(\Sigma)$ gegeben. Dann gilt:

- Aus $L \in \mathbf{L}$ folgt $\chi_L \in \text{FSPACE}(\log n)$.
- Aus $L \in \text{PSPACE}$ folgt $\chi_L \in \text{FSPACE}(n^k)$ für ein geeignetes k .

Aus diesen Resultaten folgt leicht

Satz 7 Jede nichtleere, endliche Menge ist

— \mathbf{P} -vollständig für \leq_p und

— \mathbf{L} -vollständig für \leq_{\log}

Idee: Die Reduktionsmaschine kann die gesamte Arbeit übernehmen aufgrund der vorigen Aussagen.

Offene Frage: Sind endliche Mengen in weiteren Klassen vollständig?

Diskutieren wir zwei Spezialfälle:

—**Annahme:** endliche Menge E bzgl. \leq_{\log} \mathbf{P} -vollständig

$\Rightarrow \mathbf{P} \subseteq \{A \mid A \leq_{\log} E\} \subseteq \text{DSPACE}(\log n)$

$\Rightarrow \mathbf{P} = \text{DSPACE}(\log n)$ (was vermutlich nicht gilt...)

—**Annahme:** endliche Menge E bzgl. \leq_p PSPACE -vollständig

$\Rightarrow \text{PSPACE} \subseteq \{A \mid A \leq_p E\} \subseteq \mathbf{P}$

$\Rightarrow \text{PSPACE} = \mathbf{P}$ (was vermutlich ebenfalls nicht gilt...)

Weitere Folgerungen I bei **L** und **P**:

Aus Akzeptanz der Sprache folgt Entscheidbarkeit der Sprache
in gleicher Komplexitätsklasse

Daher oft:

bei det. TM wird Entscheidbarkeit statt Akzeptanz betrachtet!

Weitere Folgerungen II

Abgeschlossenheitseigenschaften bei Komplementbildung
über die charakteristischen Funktionen:

Für $L \subseteq W(\Sigma)$ sei

$$\text{co-L} := W(\Sigma) \setminus L$$

Für eine Klasse \mathcal{K} von Problemen sei

$$\text{co-}\mathcal{K} := \{\text{co-L} \mid L \in \mathcal{K}\}$$

Lemma 8

$$\mathbf{P} = \text{co-P}$$

$$\mathbf{PSPACE} = \text{co-PSPACE}$$

$$\mathbf{L} = \text{co-L}$$

—offene Frage:

$$\mathbf{NP} \stackrel{?}{=} \text{co-NP}$$

—Nachfolgend:

$$\mathbf{NL} \stackrel{!}{=} \text{co-NL}$$

Hilfsbegriff für das Grapherreichbarkeitsproblem GAP

$G = (V, E)$ gerichteter Graph:

— *Pfad* von $x \in V$ nach $y \in V$:

Knoten-Folge x_0, \dots, x_k mit $x = x_0$, $y = x_k$ und $(x_{i-1}, x_i) \in E$

— Insbesondere: x ist Pfad von x nach x mit $k = 0$

— Existiert Pfad von x nach $y \Rightarrow y$ von x aus *erreichbar*

— G ist *stark zusammenhängend*, wenn jeder Knoten von jedem anderen Knoten erreichbar ist.

Das Grapherreichbarkeitsproblem GAP (Graph Accessibility Problem)

- **Gegeben:** gerichteter Graph $G = (V, E)$ mit $V = \{1, \dots, n\}$ und zwei Knoten $x, y \in V$.
- **Frage:** Ist y in G von x aus erreichbar?

Andere Bezeichnung: REACHABILITY

Formal mit $\Sigma := \{0, 1, (,)\}$:

GAP besteht aus allen Worten $(g)(x)(y) \in W(\Sigma)$ mit

- g Adjazenzmatrix eines Graphen $G = (\{1, \dots, n\}, E)$
- x, y binäre Darstellungen zweier Knoten von G
- in G gibt es einen Pfad von x nach y .

Lemma 9 $\text{GAP} \in \text{NL}$ sowie $\text{GAP} \in \text{P}$.

Einschub:

Komplexität von GAP (als Übung)

(nächste Vorlesung) Vollständigkeit von GAP :

Satz 10 GAP ist NL -vollständig (bzgl. \leq_{\log}).