Komplexitätstheorie

SoSe 2019

Jean Christoph Jung, Thomas Schneider

Vorstellung der weiterführenden Themen

Homepage der Vorlesung: `http://tinyurl.com/ss19-kt`

Weiterführende Themen

- NP-Vollständigkeit
- Constraint-Satisfaction-Probleme
- Schaltkreise und untere Schranken
- Komplexität des Zählens (2 Themen)
- Selbstreduktion / approximatives Zählen
- Kommunikationskomplexität
- Logarithmischer Platz und der Satz von Immerman-Szelepcsényi

Literatur ist in Stud.IP und/oder in SUUB.

NP-Vollständigkeit

Vorstellung und Beweise „klassischer“ Resultate im Kontext von P vs.NP

Satz von Ladner:

Wenn P $\neq$ NP, dann gibt es Probleme $L \in \text{NP} \setminus \text{P}$,
die nicht NP-vollständig sind (NP-intermediate Probleme).

Isomorphie-Vermutung von Berman/Hartmanis:

Alle NP-vollständigen Probleme sind paarweise p-isomorph.

(Gilt für alle bekannten NP-vollständigen Probleme.)

Satz von Mahaney:

Wenn es eine spärliche Menge gibt, die NP-vollst. ist, dann P = NP.

(„Spärlich“ = pro Wortlänge gibt es nur wenige ja-Instanzen)

Literatur: [AroraBarak09] §3.3, Aufgaben 2.30 und 6.9
[Schöning95] §15 (und dort genannte Arbeiten)

Interactive Proofs

Beweise wichtiger Begriff in der Komplexitätstheorie

z.B. Definition von NP via polynomiellen Beweissystemen

Zwei Rollen Prover = gibt den Beweis
Verifier = verifiziert den Beweis
im klassischen Fall: 1 Runde

Interactive Proofs Prover und Verifier “spielen” mehrere Runden
Prover versucht Verifier zu überzeugen

- unbeschränkte Ressourcen
- probabilistische, polynomiell Zeit-beschränkte Strategie

“überzeugen”: für alle Ja-Instanzen, sagt Verifier immer “Ja”
für alle Nein-Instanzen, sagt Verifier “Nein” mit Wkt >.5

Hauptresultat IP = PSpace
$\longrightarrow$ interaktive Beweise “mächtiger” als NP

Literatur: [Goldreich08] §9

Schaltkreise und untere Schranken

Zugang zu „P ≠ NP“ mittels Schaltkreiskomplexität:

- wurde 30 Jahre lang erforscht
- lieferte viele interessante und anspruchsvolle Resultate
- hat aber bisher nicht zum Ziel geführt

Ziel: Finde ein NP-Problem, das nicht mit einer polynomiell großen Schaltkreisfamilie entscheidbar ist

Bisher erreicht:

NP-Probleme (Familien Boolescher Funktionen),
die nicht mit eingeschränkten Schaltkreisen berechenbar sind,
z. B. Schaltkreise konstanter Tiefe oder monotone Schaltkreise

Hier sollen Überblick und Einblicke in Erreichtes/Offenes gegeben werden.

Literatur: [Vollmer99] §3.1–3.2 (Details), §3.3 (Überblick)
[Schöning95] §11 (und evtl. §12)
[AroraBarak09] (evtl. §14 für Überblick)

Komplexität des Zählens (1)

Bisher haben wir immer nur Entscheidungsprobleme betrachtet

Ist eine gegebene Formel φ erfüllbar? → ja / nein

In diesem Thema wollen wir eine Erweiterung davon betrachten

Wieviele erfüllende Belegungen hat φ ? → Entscheidungsproblem kann reduziert werden

Eigene Komplexitätsklasse **#P** ≈ “Zählvariante von NP”

grundlegende Definition, Reduktionsbegriff, vollständige Probleme

Achtung: für viele einfache Entscheidungsprobleme ist Zählen schwer

Anwendungen

Umgang mit Wahrscheinlichkeiten

Netzwerk-Zuverlässigkeit

Literatur: [Valiant79a,Valiant79b]

Komplexität des Zählens (2) — Toda’s Theorem

Wo ordnet sich Klasse #P in der klassischen Komplexitätstheorie ein?

Offensichtlich: #P “enthält” NP

#P in PSpace enthalten

PSpace

$\vdots$

Σ_3^P

Σ_2^P

$\Sigma_1^P = \text{NP}$

P

} polynomielle Hierarchie (PH)
= (wahrscheinlich) unendliche Hierarchie zwischen NP und PSpace

Satz von Toda #P enthält PH

Literatur: [Toda91], [Kozen06] Kapitel „Supplementary Lecture G“, [AroraBarak09]

Zählen versus Sampling

Uniformes Sampling

Ziel: probabilistischer Algorithmus, der alle “Lösungen” mit der gleichen Wahrscheinlichkeit ausgibt → z.B. alle erfüllbaren Belegungen

Anwendung: automatisches Testen

Beschreibe mögliche Programmeingaben durch Constraints
Teste das Programm auf zufällig gewählten Eingaben

ganz uniformes Sampling selten möglich → **approximiert uniform**

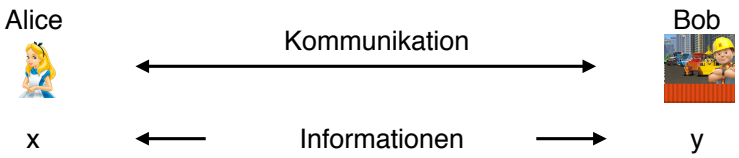
für viele Probleme sind die folgenden äquivalent:

- es existiert ein approximierender Zählalgorithmus
- es existiert ein approximiert uniformer Sampling-Algorithmus

viele = **selbstreduzierend**

Literatur: [JerrumValiantVazirani86], einführende Bemerkungen in [AroraBarak09]

Kommunikationskomplexität



Ziel **Berechnung einer Funktion f(x,y)** z.B. f(x,y)=1 gdw x=y

Frage Wieviele Bits müssen ausgetauscht werden?

Beachte die klassischen Ressourcen Zeit und Platz sind nicht relevant

Anwendungen untere Schranken für verschiedene andere Maße, z.B. Entscheidungsbäume

Beispiel-Ergebnisse:
für EQ (siehe oben) muss man alle Bits austauschen
"randomisiert" reichen O(log n) Bits

Literatur: [KushilevitzNisan06] —> UniBib oder bei uns zu leihen, [Yao79]

Log. Platz und der Satz von Immerman/Szelepcsényi

Vorstellen des überraschenden Resultats

NLogSpace = coNLogSpace (Immerman/Szelepcsényi 1988)

LogSpace alle in Platz O(log *n*) durch DTM entscheidbaren Probleme
(d. h. TM darf im Wesentlichen nur Zählerwerte speichern)

NLogSpace dasselbe, aber mit NTM
(ist im Wesentlichen Erreichbarkeit auf gerichteten Graphen)

coNLogSpace zugehörige Komplementklasse

Warum „überraschend"?
vergleiche mit NP versus coNP: das ist unbekannt; man nimmt „≠" an

Idee: raffinierter NLogSpace-Algorithmus für **Un**erreichbarkeit

Literatur: [AroraBarak09] §4.3.2
[Goldreich08] §5.3.2.3
[Kozen06] Kapitel „Lecture 4"

Wie geht's weiter?

- Themenwahl bis 30.4., Vergabe am Ende der Vorlesung
- selbstständige Bearbeitung, 1 Person pro Thema
- Wir stehen für Fragen zur Verfügung:
  zur regulären Vorlesungszeit am gewohnten Ort (MZH 6190)
- Hausarbeit (10–15 S.) und Lehreinheit durch euch Anfang Juli
- Lehreinheit: ca. 60 Min. Vorlesung, ca. 30 Min. Übung
  Block in KW 31 (29.7.–2.8.) oder KW 27/28 (1.–12.7.)?
- zwischendurch noch 2 Vorlesungen durch uns,
  vorauss. 1. Juniwoche?

Leitfaden für Seminare(!) in Stud.IP

Literatur (Lehrbücher)

[AroraBarak09] Sanjeev Arora, Boaz Barak. Computational Complexity: A Modern Approach. Cambridge Univ. Press, 2009. SUUB und Stud.IP.

[Goldreich08] Oded Goldreich. Computational Complexity: A Conceptual Perspective. Cambridge Univ. Press, 2008. SUUB (auch elektronisch).

[Kozen06] Dexter Kozen. Theory of Computation. Springer, 2006. bei uns und Stud.IP

[KushilevitzNisan06] Eyal Kushilevitz, Noam Nisan. Communication complexity. Cambridge Univ. Press, 1997. SUUB (auch elektronisch).

[Schöning95] Uwe Schöning. Perlen der Theoretischen Informatik. BI-Wiss.-Verl., 1995. SUUB und Stud.IP.

[Vollmer99] Heribert Vollmer. Introduction to Circuit Complexity: A Uniform Approach. Springer, 1999. Stud.IP

Literatur (Artikel/Konferenzbeiträge)

[JerrumValiantVazirani86] Mark R. Jerrum, Leslie G. Valiant, Vijay V. Vazirani. Random Generation of Combinatorial Structures from a Uniform Distribution. Theor. Comp. Sci. 43:169–188, 1986. Stud.IP

[Toda 91] Seinosuke Toda. PP Is As Hard As the Polynomial-Time Hierarchy. Siam J. Comput. 20(5):865–877, 1991. Stud.IP

[Valiant 79a] Leslie G. Valiant. The Complexity of Enumeration and Reliability Problems. Siam J. Comput. 8(3):410–421, 1979. Stud.IP

[Valiant 79b] Leslie G. Valiant. The Complexity of Computing the Permanent. Theor. Comp. Sci. 8:189–201, 1979. Stud.IP

[Yao79] A. C. Yao. Some Complexity Questions Related to Distributed Computing. Proc. of 11th STOC, pp. 209–213, 1979.

Weitere Quellen

- Blog „Computational Complexity“ von Lance Fortnow: `https://blog.computationalcomplexity.org/`