SS 2015

Komplexitätstheorie

Ernst W. Mayr

Fakultät für Informatik
TU München

http://www14.in.tum.de/lehre/2015SS/kt/

Sommersemester 2015

Chapter 0 Organizational Matters

- Lectures:
  - 4SWS Mon 14:15–16:00 (MI 00.08.038)
    Thu 14:00–15:45 (MI 00.08.038)
    Compulsory elective in areas Algorithms and Scientific Computing, Informatics, Bioinformatics
    Module No. IN2007
- Exercises/Tutorial:
  - 2SWS Central exercise: Tue 12:15–13:45 (03.11.018)
  - Tutor: Chris Pinkau
- Valuation:
  - 4V+2ZÜ, 8 ECTS Points
- Office hours:
  - Mon 12:00–13:00 and by appointment

- Tutorials:
  - Chris Pinkau, MI 03.09.057 (pinkau@in.tum.de)
    Office hours: Tue 14:00–15:00
- Secretariat:
  - Mrs. Lissner, MI 03.09.052 (lissner@in.tum.de)

- Problem sets and final exam:
  - problem sets are made available on Mondays on the course webpage
  - must be turned in a week later before class
  - are discussed in the tutorial
- Exam:
  - final exam: Monday, July 27, 2015, 08:30–11:30, room MW 0350
  - the final exam is closed book, no auxiliary means are permitted except for one sheet of DIN-A4 paper, handwritten by yourself
  - to pass the final exam, it is necessary to obtain at least 40% of the point total
  - probably 10 problem sets

- Prerequisites:
  - Fundamentals of Algorithms and Data Structures (GAD)
  - Discrete Probability Theory (DWT)
  - Efficient Algorithms and Data Structures (EA)
  - Randomized Algorithms
- Supplementary courses:
  - Approximation Algorithms
  - Internet Algorithmics
  - Quantum Algorithms
  - ...
- Webpage:

  http://wwwmayr.in.tum.de/lehre/2015SS/kt/

1. Planned topics for the course

1. The computational model
2. $\mathcal{NP}$ and $\mathcal{NP}$-completeness
3. Diagonalization
4. Space complexity
5. The polynomial hierarchy and alternation
6. Boolean circuits
7. (Randomized computation)
8. Interactive proofs
9. (Cryptography)
10. ...

2. Literature

- Sanjeev Arora, Boaz Barak:
  Computational Complexity — A Modern Approach,
  Cambridge University Press: Cambridge-New York-Melbourne, 2009
  http://proquest.tech.safaribooksonline.de.eaccess.ub.tum.de/9780511530753

- Giorgio Ausiello, Pierluigi Creszenzi, Giorgio Gambosi, Viggo Kann, Alberto Marchetti-Spaccamela, Marco Protasi:
  Complexity and approximation — Combinatorial optimization problems and their approximability properties,
  Springer-Verlag: Berlin-Heidelberg, 1999

- José L. Balcázar, Josep Díaz, Joaquim Gabarró:
  Structural Complexity I (and II),
  EATCS Monographs on Theoretical Computer Science, Springer-Verlag: Berlin-Heidelberg, 1995

- Christos H. Papadimitriou:
  Computational Complexity,
  Addison-Wesley Publishing Company: London-Amsterdam-New York, 1994
- Christos H. Papadimitriou, Kenneth Steiglitz:
  Combinatorial optimization: Algorithms and complexity,
  Prentice-Hall, Englewood Cliffs, NJ, 1982
- Karl Rüdiger Reischuk:
  Komplexitätstheorie — Band I: Grundlagen,
  B.G. Teubner: Stuttgart-Leipzig, 1999
  http://link.springer.com.eaccess.ub.tum.de/book/10.1007%2F978-3-322-80139-5
- Michael Sipser:
  Introduction to the Theory of Computation,
  International Edition, Thomson Course Technology:
  Australia-Canada-Mexico-Singapore-Spain-United Kingdom-United States, 2006

Ingo Wegener:
The complexity of Boolean functions,
Wiley-Teubner Series in Computer Science: Stuttgart-Chichester-New York, 1987,
http://eccc.hpi-web.de/static/books/The_Complexity_of_Boolean_Functions/

Further relevant research papers will be made available during the course.

3. Notational conventions

We use standard notation and basic concepts, as detailed e.g., in the introductory course on

Discrete Structures, IN0015

http://wwwmayr.in.tum.de/lehre/2012WS/ds/index.html.en

Chapter I The Computational Model

1. Some basic concepts

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 9–12, Cambridge University Press: Cambridge-New York-Melbourne, 2009

2. Turing Machines

2.1 The Model

2.2 Robustness

2.3 Gödel Numbers and Universal TMs

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 13–23, Cambridge University Press: Cambridge-New York-Melbourne, 2009

2.4 Universal Simulation

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 33–36, Cambridge University Press: Cambridge-New York-Melbourne, 2009

2.5 Noncomputable Functions, the Halting Problem

2.6 Deterministic Time and the Class $\mathcal{P}$

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 23–29, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Chapter II $\mathcal{NP}$ and $\mathcal{NP}$-completeness

1. The class $\mathcal{NP}$

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 37–39, Cambridge University Press: Cambridge-New York-Melbourne, 2009

1.1 Relation between $\mathcal{P}$ and $\mathcal{NP}$

1.2 Non-deterministic Turing Machines

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 39–41, Cambridge University Press: Cambridge-New York-Melbourne, 2009

2. Reducibility and $\mathcal{NP}$-completeness

3. Cook-Levin theorem

3.1 Boolean formulae and CNF

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 41–44, Cambridge University Press: Cambridge-New York-Melbourne, 2009

3.2 The Cook-Levin theorem

4. The web of reductions

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 44–50, Cambridge University Press: Cambridge-New York-Melbourne, 2009

More on "The Web of Reductions".

5. Decision versus search

6. co$\mathcal{NP}$, EXP, and NEXP

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 50–54, Cambridge University Press: Cambridge-New York-Melbourne, 2009

7. Some implications

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 55–59, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Also see

Michael R. Garey, David S. Johnson:
Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$-completeness,
W.H. Freeman and Company: New York-San Francisco, 1979

and the websites (among many others)

List of Complexity Classes (Wikipedia)
Complexity Zoo
$\mathcal{NP}$-complete Problems (Wikipedia)
$\mathcal{NP}$-completeness Columns
A Compendium of NP Optimization Problems

7.1 $\mathcal{NP}$-complete problems cannot be sparse

See

Holenstein, Thomas
Complexity Theory,
p. 4–4, Script, ETH Zürich, 2010

Shustek, Len:
An Interview with Juris Hartmanis,
Comm. ACM **58**, 4, p. 33–37, ACM Press: New York, 2015

Chapter III Diagonalization

1. Time and space hierarchy

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 63–64, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Holenstein, Thomas
Complexity Theory,
p. 2–3, Script, ETH Zürich, 2010

2. Non-deterministic time hierarchy

3. Ladner's theorem

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 65–67, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Lance Fortnow, Bill Gasarch:
Computational Complexity Blog
Two Proofs of Ladner's Theorem, 2003

4. Oracle machines and limits of diagonalization

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 68–71, Cambridge University Press: Cambridge-New York-Melbourne, 2009

On a less formal side, also see
Relativization and Multiple Choice Exams

Chapter IV Space Complexity

1. Configuration graphs

2. Some space complexity classes

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 75–78, Cambridge University Press: Cambridge-New York-Melbourne, 2009

3. PSPACE completeness

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 78–82, Cambridge University Press: Cambridge-New York-Melbourne, 2009

3.1 Savitch's theorem

3.2 PSPACE and strategies for game playing

4. $\mathcal{NL}$-completeness

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 82–88, Cambridge University Press: Cambridge-New York-Melbourne, 2009

4.1 Certificate definition of $\mathcal{NL}$: Read-once certificates

4.2 $\mathcal{NL}$ = co$\mathcal{NL}$

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 82–88, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Further references:

Larry J. Stockmeyer, Albert R. Meyer:
Word problems requiring exponential time,
Proceedings of the 5th Symposium on Theory of Computing, p. 1–9 (1973)
This paper contains some important PSPACE-completeness results.

Albert R. Meyer, Larry J. Stockmeyer:
The equivalence problem for regular expressions with squaring requires exponential space,
Proceedings of the 13th Annual Symposium on Switching and Automata Theory, p. 125–129 (1972)
This paper contains an EXPSPACE-completeness result.

And here an $\mathcal{NL}$-machine based proof for $\mathcal{NL}$ =co$\mathcal{NL}$:

Holenstein, Thomas
Complexity Theory,
p. 13–14, Script, ETH Zürich, 2010

Chapter V The Polynomial Hierarchy and Alternation

1. The class Σ_2^p

2. The polynomial hierarchy

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 91–96, Cambridge University Press: Cambridge-New York-Melbourne, 2009

3. Alternating Turing machines

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 95–96, Cambridge University Press: Cambridge-New York-Melbourne, 2009

4. Time versus alternations: Time-space tradeoffs for SAT

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 96–98, Cambridge University Press: Cambridge-New York-Melbourne, 2009

5. Defining the hierarchy via oracle machines

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 98–99, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Chapter VI Boolean Circuits

1. Boolean circuits and $\mathcal{P}_{/\mathrm{poly}}$

2. Uniformly generated circuits

3. Turing machines that take advice

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 101–105, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Leighton, F. Thomson:
Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes,
Morgan Kaufmann: San Mateo, 1992

4. $\mathcal{P}_{/\mathrm{poly}}$ and $\mathcal{NP}$: Karp-Lipton Theorem

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 106–107, Cambridge University Press: Cambridge-New York-Melbourne, 2009

5. Circuit lower (and upper) bounds

See

G.S. Frandsen and P. Bro Miltersen:
Reviewing bounds on the circuit size of the hardest functions,
IPL **95** (2005), p. 354–357

6. Non-uniform hierarchy theorem

7. Finer gradations among circuit classes

8. Circuits of exponential size

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 108–113, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Chapter VII Randomized Computation

1. Probabilistic Turing Machines

2. Some examples of PTMs

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 115–120, Cambridge University Press: Cambridge-New York-Melbourne, 2009

3. One-sided and zero-sided error: $\mathcal{RP}$, co$\mathcal{RP}$, ZPP

4. The robustness of our definitions

5. BPP $\subseteq \mathcal{P}_{/\mathrm{poly}}$

6. BPP is in $\mathcal{PH}$

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 121–128, Cambridge University Press: Cambridge-New York-Melbourne, 2009

7. Randomized reductions

8. Randomized space-bounded computation

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 129–146, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Chapter VIII Interactive Proofs

1. Interactive proofs: Some variations

1.1 Interactive proofs with deterministic verifier and prover

2. The class IP: Probabilistic verifier

3. Interactive proof for graph nonisomorphism

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 147–151, Cambridge University Press: Cambridge-New York-Melbourne, 2009

4. Public coins and AM

4.1 Simulating private coins

4.2 Set lower bound protocol

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 151–155, Cambridge University Press: Cambridge-New York-Melbourne, 2009

4.3 Some properties of $\mathcal{IP}$ and AM

4.4 Can GI be $\mathcal{NP}$-complete?

See

- Sanjeev Arora, Boaz Barak:
  Computational Complexity — A Modern Approach,
  p. 155–157, Cambridge University Press: Cambridge-New York-Melbourne, 2009

5. $\mathcal{IP}$ =PSPACE

5.1 Arithmetization

5.2 Interactive protocol for #SAT$_D$

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 157–161, Cambridge University Press: Cambridge-New York-Melbourne, 2009

Holenstein, Thomas
Complexity Theory,
p. 64–69, Script, ETH Zürich, 2010

5.3 Protocol for TQBF

6. The power of the prover

7. Multiprover interactive proofs

8. Program checking

See

Sanjeev Arora, Boaz Barak:
Computational Complexity — A Modern Approach,
p. 161–170, Cambridge University Press: Cambridge-New York-Melbourne, 2009

9. Outlook, More Topics in Complexity Theory

Among numerous other topics, see e.g. this on the GCT program:

Mulmuley, Ketan D.:
The GCT Program Toward the $\mathcal{P}$ vs. $\mathcal{NP}$ Problem,
Comm. ACM **55**, 6, p. 100–109, ACM Press: New York, 2012

For a blurb on one of the authors of the textbook for this course, see

Hyman, Paul:
An Influential Theoretician,
Comm. ACM **55**, 6, p. 24, ACM Press: New York, 2012