Fakultät für Informatik
der Technischen Universität München

Vorlesungsinhalt

Semester:	Sommersemester 2014
Vorlesung:	Komplexitätstheorie (IN2007) (4+2, 8 ECTS) (mit Übungen)
Dozent:	Prof. Dr. Ernst W. Mayr
Übungsleitung:	Chris Pinkau

Texte: Sanjeev Arora, Boaz Barak:
''Computational Complexity — A Modern Approach''
Cambridge University Press: Cambridge-New York-Melbourne, 2009

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

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 coomplexity 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/`

Vorlesungsinhalt:

0. Organizational Matters
    1. Planned Topics for the Course
    2. Literature
    3. Notational Conventions

I. The Computational Model
    1. Some Basic Concepts
    2. Turing Machines
        2.1 The Model
        2.2 Robustness
        2.3 Gödel Numbers and Universal Turing Machines
        2.4 Universal Simulation
        2.5 Non-computable Functions, the Halting Problem
        2.6 Deterministic Time and the Class $\mathcal{P}$

II. $\mathcal{NP}$ and $\mathcal{NP}$-completeness
    1. The Class $\mathcal{NP}$
        1.1 Relation between $\mathcal{NP}$ and $\mathcal{P}$
        1.2 Non-deterministic Turing Machines
    2. Reducibility and $\mathcal{NP}$-completeness
    3. Cook-Levin Theorem
        3.1 Boolean Formulae and CNF
        3.2 The Cook-Levin Theorem
    4. The Web of Reductions
    5. Decision versus Search
    6. co-$\mathcal{NP}$, EXP, and NEXP
    7. Some Implications
        7.1 Mahaney's Theorem

III. Diagonalization
    1. Time and Space Hierarchy
    2. Non-deterministic Time Hierarchy
    3. Ladner's Theorem
    4. Oracle Machines and Limits of Diagonalization

IV. Space Complexity
    1. Configuration Graphs
    2. Some Space Complexity Classes
    3. PSPACE Completeness
        3.1 Savitch's Theorem

3.2 PSPACE and Strategies for Game Playing

4. $\mathcal{NL}$-Completeness
   4.1 Certificate Definition of $\mathcal{NL}$: Read-Once Certificates
   4.2 $\mathcal{NL}$ = co-$\mathcal{NL}$

V. The Polynomial Hierarchy and Alternation

1. The Class Σ_2^p
2. The Polynomial Hierarchy
3. Alternating Turing Machines
4. Time versus Alternations: Time-space Tradeoffs for SAT
5. Defining the Hierarchy via Oracle Machines

VI. Boolean Circuits

1. Boolean Circuits and $\mathcal{P}_{/\mathrm{poly}}$
2. Uniformly Generated Circuits
3. Turing Machines that Take Advice
4. $\mathcal{P}_{/\mathrm{poly}}$ and $\mathcal{NP}$: Karp-Lipton Theorem
5. Circuit Lower (and Upper) Bounds
6. Non-uniform Hierarchy Theorem
7. Finer Gradations among Circuit Classes
8. Circuits of Exponential Size

VII. Randomized Computation

1. Probabilistic Turing Machines
2. Some Examples of PTMs
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 $\mathcal{PH}$
7. Randomized Reductions
8. Randomized Space-bounded Computation

VIII. Interactive Proofs

1. Interactive Proofs: Some Variations
   1.1 Interactive Proofs with Deterministic Verifier and Prover
2. The Class $\mathcal{IP}$: Probabilistic Verifier
3. Interactive Proof for Graph Nonisomorphism
4. Public Coins and AM
   4.1 Simulating Private Coins
   4.2 Set Lower Bound Protocol
   4.3 Some Properties of $\mathcal{IP}$ and AM
   4.4 Can GI Be $\mathcal{NP}$-complete?

5. $\mathcal{IP}$ = PSPACE
    5.1 Arithmetization
    5.2 Interactive Protocol for #SAT$_D$
    5.3 Protocol for TQBF