Computability and Complexity March 17, 2014

Summer 2014

Moritz Müller

The course offers an introduction to the theory of optimal algorithms and optimal proof systems.

1. Background from Complexity Theory

P, NP, coNP, polyomial time many-one reductions, NP-completeness of SAT (Cook's Theorem), proof thereof, simulation of Turing Machines by Boolean circuits, simulation of circuits by formulas.

2. Optimal Algorithms

Definition of optimal and almost optimal algorithms and first observations: easy subsets of hard problems, $\mathrm{E} \setminus \mathrm{P}$ contains problems with optimal algorithms, $\mathrm{E} \setminus \mathrm{P}$ contains problems without almost optimal algorithms.

3. Hard Sequences

Definition of hard sequences, observation that almost optimal algorithms do not have any, Theorem stating that coNP-complete problems without almost optimal algorithms have hard sequences.

4. Levin's Optimal Inverter

Diagonalization Lemma, Definition of inverters, Levin's Theorem.

5. Optimal SAT-Solvers

Optimal SAT-solver from Levin's inverter, associated SAT decision algorithm is length-optimal (Schnorr's result), but unlikely to be almost optimal, construction of hard instances for SAT-solvers (Gutman-Shaltiel-Ta-Shma).

6. Proof Systems

Definition propositional proof systems, proof systems, simulation order and optimality; relation to coNP versus NP (Cook-Reckhoff); optimal inverters of optimal proof systems give almost optimal decision; equivalence of the existence of optimal propositional proof systems and optimal algorithms for TAUT (Krajíček-Pudlák), generalization to paddable problems (Messner); hard sequences for proof systems.

7. Gödel Incompleteness

"T knows faster algorithms than T if and only if it proves Con_T." Gödel's 2nd Incompleteness Theorem.