6743 Review

November 29, 2007

1 Computability

Terminology: Turing machine (multi-tape, non-deterministic). Reduction (many-one). Diagonalization, dovetailing.Decidable, semi-decidable, co-semi-decidable, arithmetic hierarchy; their definitions with quantifiers. Halting problem, A_{TM}, INF. Church-Turing thesis.
Theorems: Semi-decidable $\neq$ co-semi-decidable. Their intersection is decidable. Halting problem is not decidable. There are uncountably many languages and countably many Turing machines. Simulations of multi-tape by single-tape.

2 Complexity: classes, hierarchy theorems, space vs. time

Terminology: Time and space complexity. L, P, PSPACE, EXP. Padding. Configurations and configuration graphs (and proofs using them). Hardness, completeness and examples of complete problems for all (robust) classes.
Theorems: Time and space hierarchy theorems (don't need to know exact parameters, n vs. n^2 is OK. Relations between L, P, PSPACE, EXP (know the proof idea). Know which direction the collapses go (using padding), and lintime vs. logspace problem from the assignment. Know open problems: L vs P, P vs. PSPACE.

3 Non-determinism

Terminology: Non-deterministic Turing machine, non-deterministic complexity classes NL, NP, NPSPACE, NEXP, levels of polynomial-time hierarchy. Complete problems for each of these classes (complete for NPSPACE the same as for PSPACE, TQBF; reachability for NL).Know open problems (NL vs. L, P vs. NP, NP vs. PSPACE, etc).
Theorems: Be able to prove the inclusions

$$L \subseteq NL \subseteq P \subseteq NP \subseteq PSPACE = NPSPACE$$

At this point, just know that PSPACE=NPSPACE, not the proof. Know about the non-deterministic hierarchy theorems; know which inclusions are proper because of it.

4 NP-completeness

Terminology: Definition of NP with predicates and quantifiers, NP-completeness, Clique, Independent Set, SubsetSum, Partition, Hamiltonian Path/Cycle, Satisfiability, 3SAT, 3-Colourability, $\exists SO$ logic, Horn formulas, SO Horn logic. The Cook theorem, the Fagin theorem.
Theorems: Know the steps of proofs of NP-completeness, be able to do proofs similar to the assignment. Know which problems are not NP-complete (unless P=NP). Know the general idea of the proofs of Cook's theorem and Fagin's theorem, be able to state them.

5 Space complexity and polynomial-time hierarchy

Terminology: Savitch theorem, Immerman-Szelepcsenyi theorem, reachability, transitive closure logic. Inductive counting as the technique for the proof of Immerman's theorem. TQBF. Polynomial-time hierarchy. Complete problems for NL, PSPACE, levels of PH. Collapse of PH.
Theorems: Know the statements and idea behind the proofs of Savitch theorem, and Immerman-Szelepcsenyi theorem. Be able to figure out on which level of the hierarchy a problem is by looking at the number of quantifiers.

6 Circuit complexity

Terminology: Boolean circuits, $P/poly$, AC^i, NC^i, their hierarchy. Advice Turing machines. Boolean matrix multiplication. Parity problem.
Theorems: Relationship between circuit and time/space complexity,

$$AC^0 \subsetneq TC^0 \subseteq NC^1 \subseteq L \subseteq NL \subseteq AC^1 \subseteq NC^2 \subseteq \cdots \subseteq P.$$

Parity is not in AC^0 result ($AC^0 \subsetneq TC^0$). Non-uniform circuits solving undecidable problems. Open problems: lower bounds, collapse of NC hierarchy, $P/poly$ vs. NP.

7 Randomized computation

Terminology: RP, coRP, BPP, ZPP. Notion of acceptance. Error reduction. Polynomial identity testing. Primality testing.
Theorems: Schwartz-Zippel lemma, $BPP \subset P/poly$, $BPP \subset \Sigma_2^p \cap \Pi_2^p$. Primality testing recently proven in P, before a standard example of a BPP problem. $RP \subseteq BPP$, $RP \subseteq NP$. Open problems: RP vs. coRP, BPP vs NP, BPP=P.

8 Interactive protocols

Terminology: Prover, Verifier, number of rounds, classes IP, AM, MA (in AM and MA prover sees verifier's randomness and there are 2 rounds). Class #P, problem #SAT. Zero-knowledge interactive proofs (prover tries to convince the verifier that there is a solution without giving out the solution.); statistical and computational zero-knowledge. PCP.
Theorems: Graph isomorphism and non-isomorphism as examples of interactive proof protocols; zero-knowledge proof for graph isomorphism. 3colourability has a computational zero-knowledge

proof. IP=PSPACE. $MA \subseteq AM$. Non-approximability of VertexCover problem (there is a 2-approximability, but none for a constant smaller than 2). Relation between PCP and approximation algorithms.

9 Last several theorems

Theorems: Ladner's theorem: if $P \neq NP$ then there are problems that are neither in P nor NP-complete. Schaefer's theorem: all satisfiability problems defined as conjunctions of clauses of a certain type (Horn, 2CNF, etc) are either trivial, or complete for L, NL, $\oplus$L, P or NP.