15-455: Final

May 5, 2020

Instructions

- You can use anything posted on the course website. The rest of the internet is off-limits.
- Needless to say, you may not receive help from anyone other than the course staff.
- On page one of your submission, confirm that you are in full compliance with these academic integrity rules.
- Unless explicitly stated otherwise, always justify your answer.
- Try to write up your solution in latex, it will be much appreciated. If that is impossible, scan your beautifully handwritten work and upload the scan.
- The 8 problems are of varying difficulty and are not necessarily sorted in order of increasing difficulty. You might wish to pick off the ones you find easy first.
- Upload your solution to gradescope no later than **24:00, May 5**.

Problem 1: Partitioned Turing Machines (PartTM) (10 pts.)

For this question, only consider Turing machines with a single tape. In a partitioned Turing machine the state set is partitioned into $Q = Q_R \cup Q_W \cup Q_l \cup Q_r$, the read, write, left and right states, respectively.

- Read: for a read state $p \in Q_R$ the machine scans the current tape symbol a and makes a transition into state $s(p, a)$.
- Write: for a write state $p \in Q_W$ the machine writes $w(p)$ into the current tape cell and makes a transition into state $s(p)$.
- Left: for a left move state $p \in Q_l$ the machine moves the head one cell to the left and makes a transition into state $s(p)$.
- Right: for a right move state $p \in Q_r$ the machine moves the head one cell to the right and makes a transition into state $s(p)$.

A. Show that every ordinary Turing machine $\mathcal{M}$ can be simulated by a PartTM $\mathcal{M}'$.

B. How do the two machines compare in size?

Problem 2: Difference $\mathbb{NP}$ (10 pts.)

A language L is in $D\mathbb{NP}$ if there are languages L_1 and L_2 in $\mathbb{NP}$ such that $L = L_1 - L_2$. Here is a characteristic example for a language in $D\mathbb{NP}$:

$$\mathsf{DSAT} = \{\, \varphi \# \psi \mid \varphi \in \mathsf{SAT}, \psi \in \mathsf{UNSAT} \,\}$$

where, say, $\mathsf{SAT}, \mathsf{UNSAT} \subseteq \mathbf{2}^{\star}$ codes satisfiable and non-satisfiable Boolean formulae, respectively.

A. Show that DSAT is in $D\mathbb{NP}$.

B. Show that DSAT is $D\mathbb{NP}$-complete wrto polynomial time reductions.

C. Where in the polynomial hierarchy is $D\mathbb{NP}$? Why?

For part (C) try to find the tightest upper bound you can find, but don't try to give a completeness argument.

Problem 3: Reduced Ordered Boolean Decision Diagrams (10 pts.)

We have seen that reduced ordered BDDs often provide a good implementation for Boolean formulae. A Boolean formula is symmetric if permuting the variables does not change the truth value.

A. Why is it easier to construct a small ROBDD for a symmetric formula than for arbitrary formulae?

B. What is the size of the ROBDD for the exclusive or of n variables, $x_1 \oplus x_2 \oplus \ldots \oplus x_n$?

Problem 4: Shortest Path (10 pts.)

Suppose we have a digraph $G = \langle V, E \rangle$ and two vertices s and t. A standard problem in graph theory is to determine the distance $\mathsf{dist}(s,t)$ from s to t, the length of a shortest path from s to t. For simplicity, assume $\mathsf{dist}(s,t) = \infty$ when there is no path at all. As usual, we can rephrase this function problem as a decision problem:

Problem: **Distance**

Instance: A digraph G, vertices s and t, a number k.

Question: Is $\mathsf{dist}(s,t) = k$?

A. Name a fast standard algorithm to compute $\mathsf{dist}(s,t)$ and state its time and space complexity.

B. Show that the decision problem Distance is in $\mathbb{NL}$.

Problem 5: Quadratic Residues (10 pts.)

Define the languages

$$\mathsf{QR} = \{\, a\#p \mid p \text{ prime}, a \text{ quadratic residue mod } p \,\}$$
$$\mathsf{QNR} = \{\, a\#p \mid p \text{ prime}, a \text{ quadratic non-residue mod } p \,\}$$

Here a and p are written in binary, and we assume $0 < a < p$. Recall that a is a quadratic residue if $a = x^2 \pmod{p}$ for some x.

For QNR we have an IP protocol: the verifier generates a random number r modulo p, and sends the prover either $r^2 \bmod p$ or $ar^2 \bmod p$, at random. The verifier accepts if the prover can determine which is the case.

A. Show that QR is in $\mathbb{NP}$.

B. Explain exactly why the above protocol works. Specifically, make sure that the verifier uses only polynomial time, and completeness and soundness hold.

You can use the fact that $\mathbb{Z}_p$ is a field, or the generator for $\mathbb{Z}_p^*$ mentioned in lecture 27.

Problem 6: Killing Palindromes (20 pts.)

The language $P = \{\, x\, x^{\mathsf{op}} \mid x \in \mathbf{2}^\star \,\}$ of even length palindromes is well-known not to be regular. This can be proven using Kolmogorov-Chaitin complexity (yes, that's a bit heavy-handed, but just right for a final).

Recall that given any language L and a word x, the left quotient of L, is defined as

$$x^{-1}L = \{\, y \in \Sigma^\star \mid xy \in L \,\}$$

Informally, we omit the prefix x from all words in L. For example, $a^{-1}a^\star = a^\star$ and $b^{-1}a^\star = \emptyset$. It is well-known that a language is regular iff the number of its left quotients is finite.

A. Explain left quotients for a regular language L in terms of a DFA for L.

B. Conclude that finding the length-lex minimal word in $x^{-1}L$ has constant Kolmogorov-Chaitin complexity, regardless of x.

C. Assume P is regular and concoct a contradiction by picking a nice palindrome x of length $2n$ for each n, and exploit incompressibility.

Problem 7: Integer Expressions (20 pts.)

Define an integer expression to be composed of natural numbers (written in binary as usual, x stands for the singleton $\{x\}$), and binary operations $\cup$ and $\oplus$ where

$$A \oplus B = \{\, a + b \mid a \in A, b \in B \,\}$$

Write $\mathcal{L}(E) \subseteq \mathbb{N}$ for the finite set associated with the expression E. An interval in $\mathcal{L}(E)$ is a subset $[a, b] \subseteq \mathcal{L}(E)$. We can turn this into a (slightly strange) decision problem:

Problem:	**Integer Expression Intervals (IEI)**
Instance:	An integer expression E, a number k.
Question:	Does $\mathcal{L}(E)$ have an interval of length k?

For example, $(1 \cup 2 \cup 3) \oplus (1 \cup 2 \cup 6) = \{2, 3, 4, 5, 7, 8, 9\}$ has an interval of length 4.

A. Given an integer expression E and a natural number a, show that checking whether $a \in \mathcal{L}(E)$ is in $\mathbb{NP}$.

B. Show that IEI is at level Σ_3^p of the polynomial hierarchy.

Just membership, no completeness argument is required.

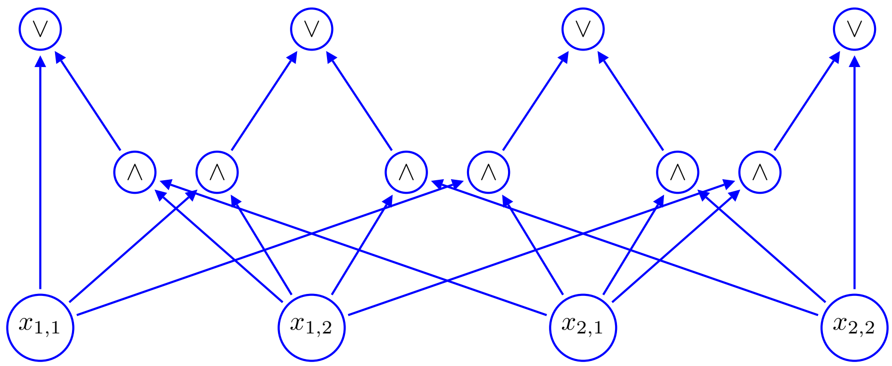
Problem 8: Transitive Closure (20 pts.)

Consider a digraph $G = \langle [n], A \rangle$ where $A \in \mathbf{2}^{n \times n}$ is the adjacency matrix (a Boolean matrix). We allow self-loops. For simplicity we will only deal with n being a power of 2.

For $n = 2$ we know a small circuit C_2 that computes the transitive closure $A^{\mathsf{tc}} \in \mathbf{2}^{n \times n}$ of A:

$$A^{\mathsf{tc}}(i, j) = 1 \iff \text{there is a path of length} \geq 1 \text{ from } i \text{ to } j$$

Note that this is the transitive closure, not the reflexive transitive closure. We will only consider Boolean circuits fan-in 2.



A. Explain this circuit in terms of matrix multiplication.

B. Explain how to construct a corresponding circuit C_n for $n = 2^k$.

C. What is the size and depth of your circuit?

Extra Credit: Argue that your circuit family is logspace-uniform.