Due: November 3th, 2017
Date: October 23, 2017

Assignment 4

You must submit it electronically to ELMS. This is a group assignment. Every group only needs to submit one solution. Group members get the same credit for the group submission.

This assignment is 7% in your total points. For the simplicity of the grading, the total points for the assignment is 70.

Problem 1 [20 pts]. Prove the following relationships among RP, CoRP and ZPP.

- (1) [10 pts] Prove that $L \in$ ZPP iff there exists an expected polynomial-time Turing machine M such that
$$Pr[M(x) = \chi_L(x)] = 1,$$
where $\chi_L(x) = 1$ if $x \in L$ and otherwise $\chi_L(x) = 0$. Note that in the lecture the definition (for ZPP) is that a probabilistic TM M that runs in polynomial time, and outputs either a correct answer (with probability at least 1/2) or "I don't know".
- (2) [10 pts] ZPP=RP$\bigcap$CoRP.

Problem 2 [20 pts]. Let A be a symmetric stochastic matrix (i.e., $A = A^T$, and every row and column of A has nonnegative entries summing up to 1). Prove that the spectral norm of A (denoted $\|A\|$) is at most 1. Note that the spectral norm of A is defined by the maximum of $\|Ax\|$ over unit vector x (i.e., $\|x\| = 1$).

Problem 3 [30 pts]. In this exercise, we show how to derive a randomness-efficient error-reduction procedure for BPP. Note that in the lecture, we only show how to do so for RP. Let G be an (n, d, λ) graph, and $\mathcal{B} \subseteq [n]$ ($|\mathcal{B}| \leq \beta n$) for some $\beta \in (0, 1)$. Let $X_1, \cdots, X_k$ be random variables denoting a $k-1$ step random walk in G from X_1, where X_1 is chosen uniformly in $[n]$.

- (1) [10 pts] For every subset $I \subseteq [k]$,
$$Pr[\forall_{i \in I} X_i \in B] \leq ((1-\lambda)\sqrt{\beta} + \lambda)^{|I|-1}.$$
- (2) [10 pts] Conclude that if $|B| < n/100$ and $\lambda < 1/100$, then the probability that there exists a subset $I \subseteq [k]$ such that $|I| > k/10$ and $\forall_{i \in I} X_i \in B$ is at most $2^{-k/100}$.
- (3) [10 pts] Use this to show a procedure that transforms every BPP algorithm A that uses m coins and decides a language L with probability 0.99 into an algorithm B that uses $m + O(k)$ coins and decides the language L with probability $1 - 2^{-k}$.
