Randomized Algorithms 2023A – Final (Take-Home Exam)

Robert Krauthgamer and Moni Naor

February 27, 2023
Due within 72 hours

General instructions. The exam has 2 parts.

Policy: You may consult textbooks and the class material (lecture notes and homework), but no other sources (like web search). You should work on these problems and write up the solutions by yourself with no help from others.

You may use without proof theorems stated in class, provided you state the appropriate theorem that you are using. As usual, assume n (or $|V|$) is large enough.

Part I (25 points)

Answer 2 of the following 3 questions. Give short answers, that sketch the proof or provide a convincing justification in 2-5 sentences, even for true/false questions.

A. Let $G = (V, E)$ be an arbitrary graph where $n = |V|$ is divisible by 3. Is it true that it must have a cut $(S, \bar{S})$ where $|S| \leq n/3$ and the cut value satisfies $\text{cut}(S, \bar{S}) > \frac{4}{9}|E|$.

   Hint: A slightly weaker bound, like non-strict inequality or a worse constant, will give you partial credit.

B. Recall that a Bloom Filter is a data structure to represent a set $S \subset U$ of size n approximately, so that for every $x \in S$ the answer is 'yes' and if $x \notin S$ is queried, then the answer is 'yes' with probability at most ϵ over the random coins used in the construction (the latter are called "false positive errors"). Recall also that we argued that the size (number of bits) of the Bloom filter has to be at least $n \log \frac{1}{\epsilon} - O(n)$.

   Consider any implementation of a data structure for approximate set membership with m bits where $m < \frac{1}{2} n \log(|U|/n)$. Suppose that there is an adversary who wants to make the data structure produce a "false positive error" and suppose that the adversary has access to the representation of the data structure in memory (knows the m bits).

   Show that for any $0 \leq \gamma < 1$ the adversary can find an element $x' \notin S$ such that the data structure answers 'yes' on x' with probability at least γ.

C. Let $G = (V, E)$ be an n-vertex tournament. Let $H \subseteq V$ be the set of vertices whose outdegree is at least $n/4$ (viewed as "high"). Aiming to find a vertex in H, we sample 1000 vertices, each one chosen independently uniformly at random from V. Is it true that with probability at least 1/8, the sampled vertices contain at least one vertex from H?

   Recall that *tournament* is a directed graph in which every pair of distinct vertices is connected by a directed edge with any one of the two possible orientations/directions (equivalently, it can be obtained by assigning a direction for each edge in a complete undirected graph).

Part II (75 points)

Answer 3 of the following 4 questions.

1. Given parameters d and ε, consider a suitable $k = O(\varepsilon^{-2}\log\frac{1}{\varepsilon})$ and a randomized linear mapping $L = G/\sqrt{k}$ where $G \in \mathbb{R}^{k\times d}$ is a random matrix of standard Gaussians.

   Let $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $W := \sum_{i<j}\|x_i - x_j\|$ (viewed as total "edge weight"). Prove that with high probability (say at least 0.9),

   $$\forall S \subseteq V, \qquad \sum_{i\in S, j\in\bar{S}} \|Lx_i - Lx_j\| \in (1\pm\varepsilon)\sum_{i\in S, j\in\bar{S}}\|x_i - x_j\| \pm \varepsilon W.$$

   (Informally, this shows that L preserves the value of every cut of these n points up to small multiplicative and additive errors.)

   Hint: To bound the LHS from above, consider for each $\{i,j\}$ whether it belongs to

   $$P^+ := \big\{\{i,j\} : \ \|Lx_i - Lx_j\| > (1+\varepsilon)\|x_i - x_j\|\big\},$$

   and use the following bound from homework (after plugging $\delta = \varepsilon$):

   $$\forall 0 \neq v \in \mathbb{R}^d, \qquad \mathbb{E}\big[\max\big\{0, |\tfrac{\|Lv\|}{\|v\|} - 1| - \varepsilon\big\}\big] \leq \varepsilon. \tag{1}$$

2. Suppose the input is split between Alice and Bob, as follows. Alice has a *directed* n-vertex graph $G = (V, E, w)$ with integral edge weights $w : E \to [n^2]$, Bob has a set $S \subset V$, and together they wish to estimate the value $v := \max\{w(S,\bar{S}), w(\bar{S}, S)\}$, where $w(A,B) := \sum_{i\in A, j\in B} w(i,j)$.

   Show that if both parties know n and $\varepsilon \in (0, 1/2)$ in advance, then Alice can send a randomized message of $O(\varepsilon^{-2} n\,\mathrm{polylog}(n))$ bits, so that Bob can output an estimate that is, with high probability, a $(1\pm\varepsilon)$-approximation to v.

   Notice: G is directed, hence we might have $w(i,j) \neq w(j,i)$.

   Hint: Sample edges based on new (symmetrized) edge weights $w'(\{i,j\}) := w(i,j) + w(j,i)$.

3. The goal of this question is to prove that for every k there is $d = d(k)$ such that every d-regular digraph (a directed graph where for all nodes the indegree and outdegree are d) contains a cycle whose length is divisible by k.

   (a) Suppose that a digraph is colored with k colors $\{0, \ldots, k-1\}$ so that for all directed edges (u, v), if u is colored i then v is colored $(i+1) \bmod k$ for some $i \in \{0, \ldots, k-1\}$. Suppose that all nodes have at least one outgoing edge. Show that the digraph contains a cycle divisible by k.

   (b) Suppose that the nodes of a d-regular digraph are colored with k colors uniformly a random and *remove* all edges that are not from color i to color $(i+1) \bmod k$ for some $i \in \{0, \ldots, k-1\}$. For each node u, what is the probability that it has at least one outgoing edge left?

   (c) Use the Local Lemma to argue that for a sufficiently large $d = d(k)$ there is a coloring where each node has outdegree at least one.

   (d) What can you say about *undirected* d regular graphs? I.e. is the statement still true?

Note: you can see a closely related proof in Alon-Spencer, Page 83 of the 3rd edition (The Probabilistic Lens: Directed Cycles). But write your proofs without relying on this.

4. Suppose that you have network of n processors arranged in a line (with one of the endpoints is designated as 'start' and the other as 'end'). Each node receives (as private input) an id - a value between 1 and n - and the goal is to check that the id's constitute a permutation.

   The protocol should work by the start node sending a message to its neighbor, which in turn send a message to its other neighbor and so on until a message reaches the 'end' node, who determines the outcome. Suggest a protocol that works with messages of length $O(\log n)$ bits and determines the correct outcome with probability at least $1 - 1/n$.

Good Luck.

THE END.