Sublinear Time and Space Algorithms 2022B – Final Assignment

Robert Krauthgamer

Due: July 21, 2022 at 11:00

General instructions: Please keep your answers short and easy to read. You can use class material (results, notation, or references) without repeating it, unless asked explicitly to do so.

Specifically for this assignment: (1) Work on the problems completely by yourself, do not discuss it with other people or search for the solution online or at other sources, this is not the intention. (2) If you use any sources other than the class material (e.g., books, online lecture notes, wikipedia, or prior knowledge), point it out, even if you happened to find the solution online — I will not deduct points, but I want to know.

Part I (36 points)

Answer all of the following 3 "short" questions. Provide a justification/proof sketch/counter-example, even for true/false questions. Keep your answer short, about 2-5 sentences; points might be deducted if the answer is too long.

A. Let $X_1, \ldots, X_n$ be pairwise-independent random signs, i.e., $\Pr[X_i = 1] = \Pr[X_i = -1] = 1/2$. Let $Y_1, \ldots, Y_n$ an independent copy of $X_1, \ldots, X_n$, i.e., drawn independently from the same distribution.

   Is it true that the $\binom{n}{2}$ random variables $\{Z_{ij} : i < j\}$, where each $Z_{ij} = X_i Y_j$, are pairwise-independent random signs?

B. Consider the frequency-vector model with deletions, where the input is a stream of additive updates to a vector $x \in \mathbb{R}^n$ whose coordinates are integers bounded by $\mathrm{poly}(n)$.

   Explain how to $(1+\epsilon)$-approximate the number of index pairs $i < j$ for which $x_i \neq -x_j$, by a streaming algorithm with storage requirement $(\epsilon^{-1} \log n)^{O(1)}$ bits.

   Remark: As done in class, do not count storage of the algorithm's random coins.

C. We saw in class a sublinear-time algorithm for approximating vertex-cover in a planar graph G with maximum degree bounded by d; specifically, it estimates $VC(G)$ within additive εn in time $T(\varepsilon, d)$ that is independent of n.

   Explain how to adapt this algorithm to the maximum independent set problem. Focus on the properties of vertex cover used in the algorithm and proof, aiming at similar guarantees (same family of graphs, same approximation, and running time that is independent of n).

   Reminder: An *independent set* in G is a set of vertices in which no two vertices are connected by an edge. It is called *maximum* if it has the largest possible size in G (in contrast, *maximal* means maximal with respect to containment).

Part II (64 points)

Answer 2 of the following 3 questions.

As done in class, do not count storage of a streaming algorithm's random coins.

1. Consider a dynamic geometric stream over $[\Delta]^d$ in the special case $d = 2$. Let $X \subset [\Delta]^2$ be the final set of points (insertions minus deletions), and to make sure it is well-defined, assume that a point can be deleted only if it was inserted earlier.

   Design a streaming algorithm that $(1 + \varepsilon)$-approximates the diameter of X.

   Hint: Assume first you are given a guess $D > 0$ that is a 2-approximation for the diameter.

   Remark: We saw in class a very simple 2-approximation algorithm for insertion-only streams; here we allow deletions and want better approximation.

2. Suppose the input is a stream of updates (insertion and deletions) of the form (i, a), i.e., each item $i \in [n]$ arrives with a *unique* identifier $a \in [n^3]$. Let X be the final set (insertions minus deletions), and to make sure it is well-defined assume that (i, a) can be deleted only if it was inserted earlier. Uniqueness means that every two items (i_1, a_1) and (i_2, a_2) in X must have $a_1 \neq a_2$, but possibly $i_1 = i_2$.

   Design a sampler for such streams, that reports $(i^*, a^*) \in X$ such that i^* is drawn uniformly from the *distinct* values in $\{i : \ (i, a) \in X\}$.

   Hint: By ignoring the identifiers, the ℓ_0-sampler seen in class will report i^* with the correct distribution. Modify this sampler to report also an identifier of i^* at some occurrence in X.

   A motivating example: The stream describes packets seen by a router, where i is the packet's destination which can be repeated, and a identifies a flow (sequence of packets from the same program) that starts/ends.

3. Consider an insertion-only frequency-vector model, i.e., the input is a stream of items from $[n]$ and $x \in \mathbb{R}^n$ is its frequency vector. Let $1 \leq T \leq n$ be a threshold given in advance (before the stream).

   Design a streaming algorithm that $O(1)$-approximates $f(x) = \sum_{i=1}^{n} \min\{x_i, T\}$ (in words, $f(x)$ sums the coordinates after truncating them at T).

   Hint: First solve the easy case $T = 1$, then consider subsampling the insertions at rate $1/T$.

THE END. Good luck!