Electronic Colloquium on Computational Complexity, Report No. 100 (2010)

A Note on Randomized Streaming Space Bounds for the Longest Increasing Subsequence Problem

Amit Chakrabarti*

Abstract

The deterministic space complexity of approximating the length of the longest increasing subsequence of a stream of N integers is known to be $\widetilde{\Theta}(\sqrt{N})$. However, the randomized complexity is wide open. We show that the technique used in earlier work to establish the $\Omega(\sqrt{N})$ deterministic lower bound fails strongly under randomization: specifically, we show that the communication problems on which the lower bound is based have very efficient randomized protocols. The purpose of this note is to guide and alert future researchers working on this very interesting problem.

1 Introduction

For a sequence σ of integers, let $\mathrm{lis}(\sigma)$ denote the length of the longest (strictly) increasing subsequence of σ. In the APPROXIMATE-LIS problem (abbreviated $\mathrm{ALIS}_{N,\varepsilon}^{M}$), we are given streaming access to a sequence σ of length N, with entries in $[M] := \{1,2,\ldots,M\}$, and must report a $(1\pm\varepsilon)$-approximation to $\mathrm{lis}(\sigma)$. The goal, as is usual in data stream algorithms [Mut03], is to minimize the space (i.e., amount of working memory) and the per-item processing time used to do so. We are concerned here primarily with the space complexity of this problem; both deterministic and randomized algorithms are of interest.

The vast majority of sublinear space streaming algorithms use randomization as a key technique and, in fact, provably need to do so [AMS99]. The ALIS problem presents one of the very few instances where (1) a natural problem has a sublinear space deterministic streaming algorithm, (2) randomization is not known to provide any extra space savings, and (3) randomization *could* provide an exponential improvement, based on current knowledge. We believe that this makes the randomized space complexity of ALIS an extremely interesting theoretical question about data stream algorithms.

A sublinear upper bound for ALIS was given by Gopalan et al. [GJKK07], who showed the following.

Theorem 1.1. *There is a deterministic $O(\sqrt{N/\varepsilon}\cdot\log M)$-space streaming algorithm for $\mathrm{ALIS}_{N,\varepsilon}^{M}$.*

An essentially matching lower bound was then given by Gál and Gopalan [GG07] and also — independently and via a different argument — by Ergün and Jowhari [EJ08]. These lower bounds applied only to deterministic algorithms and used reductions from certain communication problems on which suitable "direct sum" theorems could be proven. In this note, we show that these techniques do not generalize to give randomized streaming lower bounds, because the underlying communication problems *do* have randomized protocols with cost exponentially lower than the best deterministic protocol.

2 Preliminaries

For a communication problem f, let $\mathrm{D}^{\max}(f)$ denote the maximum number of bits sent by any *single* player in a deterministic protocol that computes f, minimized over all such protocols. Let $\mathrm{R}^{\max}(f)$ denote the analogous quantity for constant-error randomized protocols.

*Dartmouth College. ac@cs.dartmouth.edu. Supported in part by NSF Grants CCF-0448277 and IIS-0916565.

ISSN 1433-8092

In the HIDDEN-INCREASING-SEQUENCE problem (abbreviated $\mathrm{HIS}^m_{t,n,k}$), the input is a $t \times n$ matrix $X = \{x_{ij}\}_{i\in[t],j\in[n]}$ with entries in $[m]$, with the promise that each column X_j of X satisfies one of the following:

1. The column is non-increasing, i.e., $\mathrm{lis}(X_j) = 1$.
2. The column has a "long" increasing subsequence, i.e., $\mathrm{lis}(X_j) \ge k$.

This input is divided amongst t players, named $\mathrm{PLR}_1, \ldots, \mathrm{PLR}_t$, with player i receiving the ith row of X. The goal is to compute the predicate $\mathrm{HIS}^m_{t,n,k}(X) := \bigvee_{j=1}^n (\mathrm{lis}(X_j) \ge k)$, in the following model of communication: PLR_1 sends a (private) message to PLR_2, who then sends a message to PLR_3 and so on, until we reach PLR_t, who then announces the output.

Let σ_X denote the length-(tn) sequence, with elements in $[tm]$, obtained by applying the mapping $x_{ij} \mapsto (j-1)m + x_{ij}$ to the entries of X and reading off the result in row-major order. It is easy to see that

$$\mathrm{HIS}^m_{t,n,k}(X) = 0 \implies \mathrm{lis}(\sigma_X) = n, \quad \text{and} \quad \mathrm{HIS}^m_{t,n,k}(X) = 1 \implies \mathrm{lis}(\sigma_X) \ge n+k-1. \tag{1}$$

This was first formally observed by Gopalan et al. [GJKK07], and it immediately implies the following.

Lemma 2.1 (Lemma 4.4 of [GJKK07])**.** *The deterministic and randomized streaming space complexities of $\mathrm{ALIS}^M_{N,\varepsilon}$, with $M = tm$, $N = tn$ and $\varepsilon = (k-1)/n$, are at least $\mathrm{D}^{\max}(\mathrm{HIS}^m_{t,n,k})$ and $\mathrm{R}^{\max}(\mathrm{HIS}^m_{t,n,k})$, respectively.*

Thus, a natural approach to establishing space lower bounds on ALIS is to prove communication complexity lower bounds for HIS. Using this approach, Gál and Gopalan proved a tight determinstic lower bound:

Theorem 2.2 (Theorems 1.1 and 4.1 of [GG07])**.** *Let H_b denote the binary entropy function. Then, we have*

$$\mathrm{D}^{\max}(\mathrm{HIS}^m_{t,n,k}) \;\ge\; n\left(\left(1-\frac{k}{t}\right)\log\left(\frac{m}{k-1}\right) - H_b\left(\frac{k}{t}\right)\right) - \log t\,.$$

In particular, setting $k-1 = t/2 = \varepsilon n$, we have $\mathrm{D}^{\max}(\mathrm{HIS}^m_{t,n,k}) = \Omega(n\log(m/\varepsilon n))$. Combining this bound with Lemma 2.1 shows that the deterministic space complexity of $\mathrm{ALIS}^M_{N,\varepsilon}$ is $\Omega(\sqrt{N/\varepsilon}\cdot\log(M/\varepsilon N))$.

In fact, they also generalized this theorem to apply to multi-pass, but still deterministic, streaming algorithms by extending the communication lower bound to multi-round protocols. Our concern here is with a different potential generalization: can one generalize Theorem 2.2 to *randomized* protocols, and thus, randomized streaming algorithms? Our main result is a negative one, showing that this is not possible.

3 A Randomized Communication Upper Bound

Theorem 3.1 (Main Theorem)**.** *We have*

$$\mathrm{R}^{\max}(\mathrm{HIS}^m_{t,n,k}) \;=\; O\left(\frac{nt\log m}{k^2}\right).$$

In particular, for the setting $k = \Theta(t) = \Theta(\varepsilon n)$, which was used for the lower bound in Theorem 2.2, we have $\mathrm{R}^{\max}(\mathrm{HIS}^m_{t,n,k}) = O(\varepsilon^{-1}\log m)$.

Proof. Let $r = 2(t-1)/(k-1)$. Consider the following protocol. Each player goes through a receive-and-compute phase (skipped by PLR_1) followed by a transmit phase (skipped by PLR_t). In the transmit phase, PLR_i chooses a subset $J_i \subseteq [n]$ of size $|J_i| = \lceil 2n/(k-1)\rceil$ uniformly at random from amongst all such subsets. He then sends to PLR_{i+1} the following data.

1. The set $S_i = \{x_{ij} : j \in J_i\}$; for each $j \in J_i$, we say that PLR_i *samples* column j.
2. The sets S_h for $\max\{1, i-r+1\} \le h < i$, which he obtains from PLR_{i-1}.

In the receive-and-compute phase (which precedes the transmit phase), PLR_i receives from PLR_{i-1} the sets S_h, for all $h \in H_i := \{h : \max\{1, i-r\} \le h < i\}$. He checks whether the following condition holds:

$$\exists h \in H_i \ \exists j \in J_h : \ x_{hj} < x_{ij} \,. \tag{2}$$

He can do so because x_{hj} is available to him from the message he receives and x_{ij} is within the part of the input he knows to begin with. If (2) holds, he terminates the protocol and outputs 1. Notice that he is correct to do so, because he has discovered that $\text{lis}(X_j) > 1$, which, by the promise, means that $\text{lis}(X_j) \ge k$. Otherwise, if (2) does not hold, he moves on to the transmit phase as described above.

If the protocol reaches PLR_t, and no player (including himself) has output 1 in their receive-and-compute phase, then he outputs 0. This completes the description of the protocol. Clearly, one can tweak it so that the output is always announced by PLR_t, as in our definition.

We now argue that this protocol is correct. Suppose that $\text{HIS}^m_{t,n,k}(X) = 0$. Then (2) never holds, and the protocol always reaches PLR_t, who correctly outputs 0.

Suppose that $\text{HIS}^m_{t,n,k}(X) = 1$, and suppose that the jth column of X contains a "hidden increasing subsequence" $\langle x_{ij} \rangle_{i \in I}$, where $I \subseteq [t]$ and $|I| = k$. Call the player PLR_i *critical* if the following condition holds:

$$i \in I \ \wedge \ \left(\exists i' \in I : \ 0 < i' - i \le r \right) . \tag{3}$$

Also, in this case, call $\text{PLR}_{i'}$ the *follower* of PLR_i, where i' is the minimum integer such that $0 < i' - i \le r$.

A straightforward estimation argument shows that there exist at least $(k-1)/2$ critical players. Notice that if a critical player samples column j, then his follower correctly announces the output to be 1. Thus, the probability that the protocol fails to output 1 is at most

$$\Pr[\text{no critical player samples column } j] \ \le \ \left(1 - \frac{2}{k-1}\right)^{(k-1)/2} \ \le \ e^{-1} \,.$$

Finally, note that in order to achieve this constant error probability, each player had to send at most $r \cdot \lceil 2n/(k-1) \rceil$ entries of X, which required $O((nr/k) \cdot \log m) = O((nt/k^2) \cdot \log m)$ bits. $\square$

4 Concluding Remarks

We remark that a similar randomized communication upper bound can be shown to hold for the slightly different communication problem used by Ergün and Jowhari [EJ08] in their alternate proof of the deterministic lower bound for ALIS.

These protocols raise an obvious question: does the idea extend to give a polylogarithmic-space streaming upper bound for ALIS? We leave this question open.

References

[AMS99] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999. Preliminary version in *Proc. 28th Annu. ACM Symp. Theory Comput.*, pages 20–29, 1996.

[EJ08] Funda Ergün and Hossein Jowhari. On distance to monotonicity and longest increasing subsequence of a data stream. In *Proc. 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 730–736, 2008.

[GG07] Anna Gál and Parikshit Gopalan. Lower bounds on streaming algorithms for approximating the length of the longest increasing subsequence. In *Proc. 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 294–304, 2007.

[GJKK07] Parikshit Gopalan, T. S. Jayram, Robert Krauthgamer, and Ravi Kumar. Estimating the sortedness of a data stream. In *Proc. 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 318–327, 2007.

[Mut03] S. Muthukrishnan. Data streams: Algorithms and applications. In *Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, page 413, 2003.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de