CS 15-851: Algorithms for Big Data	Spring 2024
Lecture 8 (Part two) — Mar 14	
Prof. David Woodruff	*Scribe: Aashiq Muhamed*

1 One-Way Communication Complexity For The Indexing Problem

In this section, we look at the one-way communication complexity for the Index problem. We examine the scenario where Alice possesses a binary string $x \in \{0,1\}^n$ and is tasked with sending a singular message M to Bob. Given M and an index $j \in [n]$, Bob's objective is to accurately determine the value of x_j with a success probability of at least $\frac{2}{3}$. We note that this probability threshold is determined by the randomness of the process, applicable universally across all strings x and indices j, expressed as $\forall x, \forall j, \Pr_{\text{randomness}}[\text{Bob correctly identifies } x_j] \geq \frac{2}{3}$.

Our goal is to establish that the minimum amount of information that must be exchanged to address this problem effectively is $\Omega(n)$ bits. In our analysis, we use a uniform distribution μ over the set of strings $X \in \{0,1\}^n$, and compare Bob's prediction X'_j against the actual value X_j, acknowledging that the probability of correctness $\Pr\left[X'_j = X_j\right] \geq \frac{2}{3}$. This scenario prompts us to question the information about X_j that can be inferred from M. Leveraging Fano's inequality, given the independence of X_j and X'_j conditioned on M as illustrated by the Markov chain $X \to M \to X'$, we derive:

$$\begin{aligned}
H(X_j \mid M) &\leq H(P_e) + P_e \log_2(|X| - 1) && \text{Application of Fano's Inequality} \\
&\leq H\left(\frac{1}{3}\right) + \frac{1}{3}\log_2(2-1) && \text{Binary case, } |X| = 2, P_e = \frac{1}{3} \\
&< 1, && (1)
\end{aligned}$$

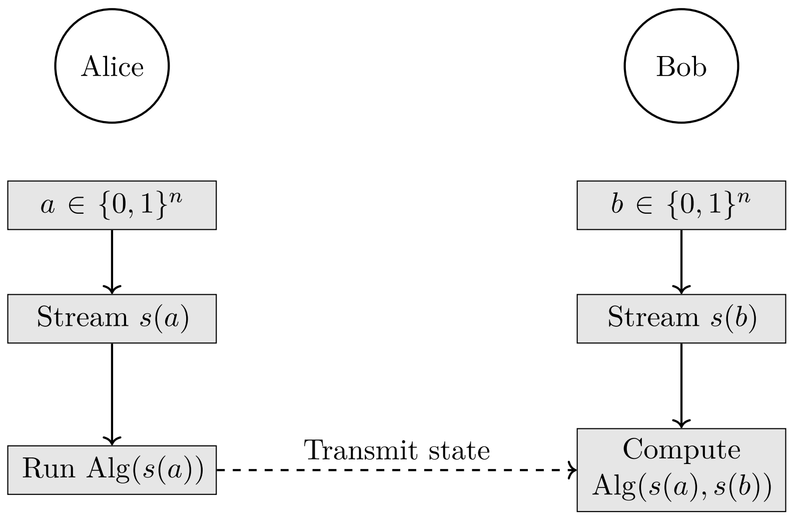
indicating that the message M indeed reveals some information about X. Further, exploring the mutual information $I(M;X)$, with M as a variable dependent on X, and X uniformly distributed, we apply the chain rule to obtain a lower bound. Denoting $X_{<i}$ as the sequence of bits preceding the i-th bit, and considering X as a binary sequence, we find:

$$\begin{aligned}
I(X;M) &= \sum_i I(X_i; M \mid X_{<i}) && \text{Independence of bits in the sequence} \\
&= n - \sum_i H(X_i \mid M, X_{<i}) && \text{Definition of Mutual Information} \\
&\geq n - \sum_i H(X_i \mid M) && H(X_i \mid M, X_{<i}) \leq H(X_i \mid M) \\
&\geq n - nH\left(\frac{1}{3}\right), \\
&= \Omega(n), && \text{By Fano's Inequality (1)} \qquad (2)
\end{aligned}$$

thus establishing the mutual information as at least $\Omega(n)$ bits. Relating this to the communication complexity represented by $|M| = \ell$ bits, which delineates 2^ℓ potential states, and recognizing $H(M) \leq \log_2(2^\ell) = \ell$, we deduce $|M| \geq H(M) \geq I(X;M) = \Omega(n)$, proving that the size of Alice's message must span at least $\Omega(n)$ bits.

2 Typical Communication Reduction

In our efforts to establish lower bounds for computational problems, it's a common strategy to reduce the problem in question to another problem like the Index problem to compute a lower bound. Below is a visual framework depicting this reduction process.



Following this process:

1. Alice runs a streaming algorithm on $s(a)$, then transmits the state of Alg$(s(a))$ to Bob.
2. Bob computes Alg$(s(a), s(b))$.
3. If Bob successfully solves $g(a, b)$, it implies that the space complexity of Alg is at least the one-way communication complexity of g.

3 Case Study: Identifying Distinct Elements

We are interested in calculating the total count of distinct elements within a series $a_1, a_2, \ldots, a_m \in [n]$. The aim is to illustrate that the minimum required communication complexity for solving this challenge effectively is $\Omega(n)$ bits, establishing a foundational lower bound.

We approach this by reducing the current problem of enumerating distinct elements to the Index problem that we are familiar with. Briefly revisiting the Index problem, it involves Alice holding a binary sequence $x \in \{0,1\}^n$, and Bob having an index $i \in [n]$, with the objective for Bob being to determine the truth value of $x_i = 1$.

The methodology for the reduction is described as follows: Let $s(a)$ represent the sequence $i_1, i_2, \ldots, i_r$, including each i_j when $x_{i_j} = 1$. We define $s(b) = i$, and analyze what happens for each of x_i:

$$x_i = \begin{cases} 0 & \text{if } \mathrm{Alg}(s(a), s(b)) = \mathrm{Alg}(s(a)) + 1 \\ 1 & \text{otherwise} \end{cases}$$

In instances where $\text{Alg}(s(a), s(b)) = \text{Alg}(s(a)) + 1$, this indicates that the inclusion of $s(b)$ with the stream introduces an extra unique element. If $x_i = 1$, suggesting i is included in $s(a)$, then the inclusion of $s(b)$ signifies that i is no longer recognized as a new distinct element. On the other hand, if $x_i = 0$, the count of unique elements is incremented by one.

This reduction process implies that the space complexity for Alg is necessarily at least equivalent to the one-way communication complexity of the Index problem.

4 Strengthening Index: Augmented Indexing Approach

In this enhancement of the Index problem, we augment the information available to Bob by providing him access to the sequence of bits up to x_{i-1}. Bob's objective remains unchanged; he aims to ascertain the value of x_i based on the information relayed by Alice.

Participant	Alice	Bob
Information	$x \in \{0,1\}^n$	$i \in [n]$ and $x_1 \ldots x_{i-1}$
Stream	$s(a)$	$s(b)$

We assert that this modified problem still adheres to a lower bound of $\Omega(n)$, following a reasoning akin to the original Index problem. The proof uses the mutual information concept, presented as follows:

$$\begin{aligned}
I(X;M) &= \sum_i I(X_i; M|X_{<i}) && \text{Chain rule} \\
&= \sum_i \left(H(X_i|X_{<i}) - H(X_i|M, X_{<i})\right) && \text{Definition of Mutual Information} \\
&= n - \sum_i H(X_i|M, X_{<i}) && \text{Independence of } X_i \text{ and } X_{<i} \text{ and uniform } X_i
\end{aligned}$$

For this analysis, we leverage the Markov Chain $X \to X_{<i}, M \to X_i'$ in conjunction with Fano's Inequality, leading to $H(X_i|M) \leq H(\delta) + \delta \log_2(2-1) = H(\delta)$. Continuing with the proof of the lower bound:

$$\begin{aligned}
I(X;M) &\geq n - H(\delta)n \\
&\geq n(1 - H(\delta))
\end{aligned}$$

Consequently, this approach enables us to establish a general lower bound on the communication complexity (CC) for the Augmented Index problem:

$$CC_\delta(\text{Augmented Index}) \geq I(M;X) \geq n(1 - H(\delta))$$

5 Establishing a $\log(n)$ Bit Lower Bound for Norm Estimation

In this example, we explore the case where Alice's input to the Augmented Index problem is restricted to $\log(n)$ bits, to define a lower bound for norm estimation tasks. A fundamental problem in streaming algorithms involves managing a counter through a sequence of increments and decrements. The pivotal question we examine is whether it's feasible to estimate this counter within a factor of 2. Although directly reporting such an approximation requires only $\log(\log(n))$ bits storage, we demonstrate that the actual communication complexity of approximation transcends the mere space to store the counter.

Alice initiates this process by constructing a vector v, which contains a single non-zero coordinate represented by $\sum_i^{\log(n)} 10^j x_j$. This setup implies that estimating the norm of vector $v = (c, 0, 0, \ldots, 0)$ is synonymous with approximating the value of c across any p-norm $\|v\|_2, \|v\|_1, \ldots, \|v\|_p$. Following this, Alice transmits to Bob the algorithm's state after incorporating vector v into the data stream.

Bob, holding an index $i \in [\log(n)]$ along with subsequent bits $x_{i+1}, x_{i+2}, \ldots, x_{\log(n)}$, constructs the vector $w = \sum_{j>i} 10^j$. The algorithm's output after adjusting vector v by vector w, $\text{Alg}(v-w) = \text{Alg}\left(\sum_{j \le i} 10^j x_j\right)$, can be used to determine the value of x_i:

$$x_i = \begin{cases} 1 & \text{if } \text{Alg}(v-w) \ge \frac{10^i}{2} \\ 0 & \text{otherwise} \end{cases}$$

For instances where $x_i = 0$, the algorithm's output, $\text{Alg}(v-w)$, is constrained by the sum $1+10+100+\ldots+10^{i-1}$, which is less than $\frac{2}{9} \cdot 10^i$. Conversely, if $x_i = 1$, then $\text{Alg}(v-w) \ge \frac{10^i}{2}$. This separation establishes a gap between the bounds, even in the factor of 2-approximation framework. Thus the norm approximation problem can be reduced to the index problem and estimating norms requires a lower bound of $\log(n)$ bits.

6 $\frac{1}{\varepsilon^2}$ Bit Lower Bound for Norm Estimation

In this analysis, we aim to demonstrate a $\frac{1}{\varepsilon^2}$ bit lower bound for norm estimation by connecting the Index problem with the Gap Hamming Distance problem, following the methodology outlined in (1). The Gap Hamming Distance problem is equivalent to the norm estimation problem for binary streams.

- **Hamming Distance** $\Delta(x, y)$: Counts the number of positions i at which the corresponding symbols x_i and y_i differ.

- **Gap Hamming Problem**: Given two strings of length n, we're assured that either $\Delta(x, y) > \frac{n}{2} + 2\varepsilon n$ or $\Delta(x, y) < \frac{n}{2} + \varepsilon n$. The objective is to ascertain which condition holds.

- **Public Coin**: Denotes a series of bit vectors $r^1, \ldots, r^t$, or equivalently, a $t \times t$ matrix of random bits where row i is represented by r^i.

Alice is provided a bitstream x of length $t = O\left(\frac{1}{\varepsilon^2}\right)$, denoted $a = \{0,1\}^t$. Bob receives an index i with $b = \{0,1\}^t$. Using the public coin, a and b are defined as as follows:

$$a_k = \text{Majority}_{j \text{ where } x_j = 1} r_j^k,$$
$$b_k = r_i^k.$$

Here, b simply uses the i-th column from the public coin matrix, whereas a aggregates columns for which $x_j = 1$, and then determines the majority bit for each row. If $x_i = 0$, a and b remain unrelated due to the exclusion of the i-th column in a's construction, leading to an expected Hamming distance $\mathbb{E}[\Delta(a,b)] = \frac{t}{2}$. This is because each bit matches with a $\frac{1}{2}$ probability. Conversely, if $x_i = 1$, a and b become dependent. Examining the likelihood for a bit (1 or 0) to represent the majority, and considering the weight of a vector as its count of 1s, let k denote the weight of x. The probability that the majority bit is equal to the first bit can be written after approximating the sum of binomial coefficients as,

$$\begin{aligned}\Pr\left[\text{majority of } Z_1, \ldots, Z_k = Z_1\right] &\approx \frac{1}{2} + \Theta\left(\frac{1}{\sqrt{k}}\right) \\ &\approx \frac{1}{2} + \Theta\left(\frac{1}{\sqrt{t}}\right) \\ &\approx \frac{1}{2} + \Theta(\varepsilon).\end{aligned}$$

This leads us to an adjusted expected Hamming distance:

$$\begin{aligned}\mathbb{E}[\Delta(a,b)] &= \frac{t}{2} - \Theta(\varepsilon t) \\ &= \frac{t}{2} - \Theta\left(\frac{1}{\varepsilon}\right).\end{aligned}$$

In terms of x_i's value, this can be written as:

$$\mathbb{E}[\Delta(a,b)] = \frac{t}{2} - x_i\sqrt{t}.$$

The gap across the two scenarios varies by a factor of $(1+\varepsilon)$ meaning that we can use Gap Hamming Algorithm to resolve Indexing. As t is set to $\Theta\left(\frac{1}{\varepsilon^2}\right)$, this derivation implies a $\frac{1}{\varepsilon^2}$ bit lower bound for the Gap Hamming problem and therefore the norm estimation problem.

References

[1] Thathachar S Jayram, Ravi Kumar, and D Sivakumar. The one-way communication complexity of hamming distance. Theory of Computing, 4(1):129-135, 2008.