Massive Data Algorithmics

The Streaming Model

Lecture 17: Communication Complexity

Communication Game

- There are two parties to the communication game, namely Alice and Bob.
- Alice's input $x \in X$ and Bob's input $y \in Y$.
- We want to compute $f(x,y), f : X \times Y \to Z$ when $X = Y = [n], Z = 0, 1$.
- For example, consider $f(x,y) = x + y \pmod 2$.
- In this example, Alice does not have to send the whole input x using $\log n$ bits; instead she can send x mod 2 to Bob using just 1 bit. Bob now can compute $f(x,y)$. However, only Bob knows the answer. Bob can choose to send the result to Alice. But in this model, it is not required that all the players should know the answer.
- We are not concerned about the memory usage, but we try to minimize the cost of communication between Alice and Bob.

Equality Problem

- Given $X = Y = \{0,1\}^n$ and $Z = \{0,1\}$
- $\mathrm{EQ}(x,y) = 1$ if $x = y$. Otherwise, $\mathrm{EQ}(x,y) = 0$
- We are in the one-way model where messages are sent in one direction.
- For symmetic fucntions, it does not mater who sends the message to whom.

Theorem. Alice must send n bits in order to solve EQ in the one-way model, i.e.

$$D^{\rightarrow}(EQ) \geq n$$

Lower bound

Proof.

- Suppose Alice sends $< n$ bits to Bob.
- Then the number of different messages she might send $\leq 2^1 + 2^2 + \cdots + 2^{n-1} = 2^n - 2$ but Alice can have upto 2^n inputs.
- Using the pigenhole principle, there exist two input $x \neq x'$ such that alice sends the same message α on input x and x'.
- Let $P(x,y)$ be Bob's output when the input is (x,y).
- We should have $P(x,y) = \mathrm{EQ}(x,y)$
- $P(x,x) = P(x',x)$ as Bob sees the message α for both x and x'.
- $\mathrm{EQ}(x,x) = 1$ and $\mathrm{EQ}(x',x) = 0$ which contradict $P(x,y) = \mathrm{EQ}(x,y)$.

Randomization Algorithm

Theorem. Using randomness, we can compute EQ function with error probability $\leq 1/3$ in the one-way model with message size $O(\log n)$ bits.

Proof.

Algorithim:

- Alice picks a random prime $p \in [n^2, 2n^2]$.
- Alice sends $(p, x \bmod p)$ using $O(\log n)$ bits.
- Bobs check if $y \bmod p = x \bmod p$, outputs 1 if true and 0 otherwise.

Randomization Algorithm

Probability error.
If $\mathrm{EQ}(x,y) = 1$, output is correct. If $\mathrm{EQ}(x,y) = 0$, then it has an error iff $p|x-y$. We can bound the probability error as follows.

- $|x-y| = p_1^{\alpha_1} \cdots p_t^{\alpha_t}, |x-y| \le 2^n \rightarrow t \le n$
- $\Pr(error) \le \frac{t}{\#\text{ primes in } [n^2, 2n^2]}$.
- $\#$ primes in $[1,N]$ is about $\frac{N}{\ln N}$.
- So $\#$ primes in $[n^2, 2n^2]$ is about $\frac{2n^2}{\ln(2n^2)} - \frac{n^2}{\ln(n^2)} \ge \frac{0.9n^2}{2\ln n}$
- $\Pr(error) \le \frac{n}{(0.9n^2)/(2\ln n)} = \frac{2\ln n}{0.9n} \le \frac{1}{3}$

Reduction

Theorem. Suppose there exists a deterministic or randomized streaming algorithm to compute $f(x,y)$ using s bits of memory, then $D^{\rightarrow}(f) \leq s$.

Proof.

- Alice runs the algorithm on her part of the stream, sends the values in the memory (s bits) to Bob, and he uses these values along with his part of the stream to compute the output.

Corollary. If $D^{\rightarrow}(f(x,y)) \geq s$ in the one-way model of the communication complexity, then any streaming algorithm computing $f(x,y)$ must use s bits of memory.

Index Problem

- Given $X = \{0,1\}^n, Y = [n]$ and $Z = \{0,1\}$
- $\mathrm{Index}(x,j) = x_j =$ j-th bit of x.
- For example, $\mathrm{Index}(1100101,3) = 0$.

Theorem. $D^{\rightarrow}(\mathrm{Index}) \geq n$

Proof. Use the pigenhole principle to prove the theorem.

Majority Problem

- Input: the stream $\sigma = < a_1, \cdots, a_m >$ where $a_i \in [n]$
- Output: if $\exists j : f_j > m/2$, output j. Otherwise output null.

Let $s(n,m)$ be the minimum size of the memory used by any streaming algorithm

Reducing Index to Majority

- Given an instance (x,j) of Index.
- We construct streams σ and π of length n each as follows. Let A be the streaming algorithm for Majority.
  - Alice's input x is mapped to $\sigma = a_1, a_2, \cdots, a_n$, where $a_i = 2(i-1) + x_i$.
  - Bob's input j is mapped to $\pi = b, b, \cdots, b$, where b occures n times and $b = 2(j-1)$.
  - Alice and Bob communicate by running A on $\sigma \cdot \pi$.
  - If A says "no majority", then output 1, else output 0.
- Therefore, $s(2n,2n) \geq D^{\rightarrow}(\text{Index}) = n$ or equivalenty $s(n,n) \geq n/2$.
- In genearl we can show $s(n,m) \geq \min(n,m)/2$.
- Easy to see with about $\min(n,m)$ words $(\min(n\log n, m\log m))$ of memory we can solve the majority problem.

References

- **Data Stream Algorithms** (Chapter 15)
  Lecture notes by A. Chakrabbarti and D. College