Algorithms & The Order of the Stream

S. Guha
UPENN

joint work with A. McGregor

Data Stream Model

- Adversary controls the order of the input.
  - Upper bound statements are very powerful
  - Few things have nice upper bounds – response of boring to paranoia from non-theorists

Random Order Model

- Worst case over the distribution
- Assumes that once the input is fixed, any permutation is equally likely.
- Average case model
- Random order generalizes assumptions such as Zipf, Gaussian, etc

Why 1. A Classic Model

- Munro, Paterson `80
  - Exact Algorithms
  - $O(n^{1/p})$ space using p passes
  - $O(n^{1/(2p)})$ space for random order streams
  - Open Problem: O(log log n) passes using $\log^{O(1)} n$ space.

Why 2. Power of Adversary

- Genie in the network hates you.
- All packets are delivered at exactly the wrong point of time
- Adversary rearranges after looking at the full input
- Limited Adversaries …
- Say for a network the sum of the queue sizes …

Why 3. Natural Model

- Random by
  a) Definition: iid samples from a distribution. More on this later.
  b) Semantics: (Firstname, Salary)
  c) Design: Backup samples in disc.

Why 4. Algorithm Development

- Discrepancy Method
- Assume random order
- Develop algorithm
- Simulate random order

    - Meyerson ’01 (showed both)
        - used in Charikar, Panigrahy, O’callaghan ‘03)
    - Chang, Kannan ’06; Guha, McGregor ’07
    - Guha, McGregor, Venkatasubramanian ’06; Bhuvnagari, Ganguly ’06, Chakrabarti, Cormode, McGregor ’07.

Few Results

- Demaine, Lopez-Oritz, Munro, 02

- Takers?
- Has to be a permutation invariant function.

3a) Brave New World

- Assume you are trying to estimate some property of a distribution
    - Why did you want the median anyway?

- You need a bunch of samples (VC Theory)
- Do you need to **store** all?

Space as Precision

- Find the CDF=½ point.

- ε^{-2} samples give § ε guarantee
- Space S) § n/S we get ½§ 1/S

- Does not improve if n ! 1
- Sad.

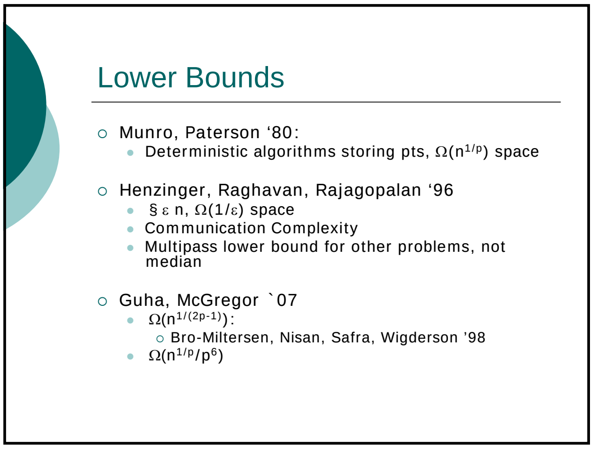
Looking Ahead

- Consistent Estimation
- Also Exchangeability and deFinetti

- If space S gave § (pn) $\log^{O(1)}$ n/S
- As n ! 1 then § term ! 0

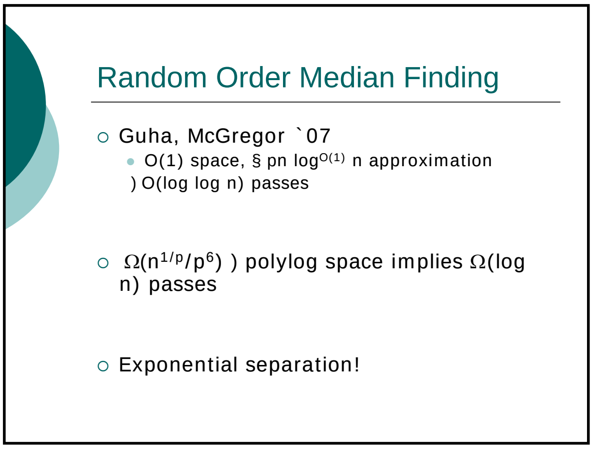
Adversarial Order Upper Bounds

- Munro Paterson
    - $n^{1/p}$ space for p passes exact
- Manku, Rajagopalan, Lindsay `98
    - ε-^{1} $\log^2 n$ for § ε n
- MRL 99, Greenwald & Khanna `01
    - ε^{-1} log n
- 1/ε points define a coreset in 1D
    - With a log n loss …
- Sampling ε^{-2} log n for § ε n
- Consequence
    - $\varepsilon^{-1/p}$ for § ε n in p passes



Lower Bounds

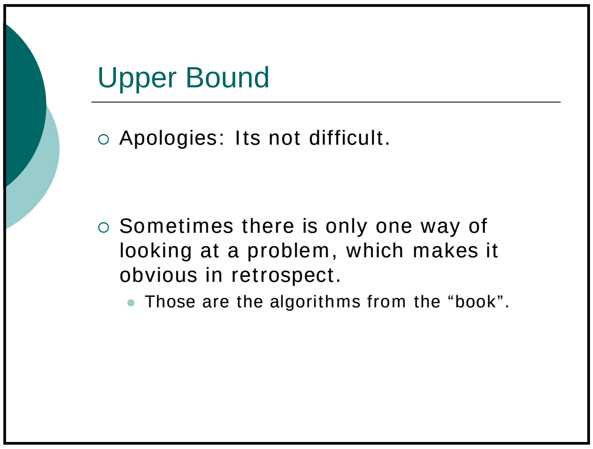
- Munro, Paterson '80:
  - Deterministic algorithms storing pts, $\Omega(n^{1/p})$ space
- Henzinger, Raghavan, Rajagopalan '96
  - § ε n, $\Omega(1/\varepsilon)$ space
  - Communication Complexity
  - Multipass lower bound for other problems, not median
- Guha, McGregor `07
  - $\Omega(n^{1/(2p-1)})$:
    - Bro-Miltersen, Nisan, Safra, Wigderson '98
  - $\Omega(n^{1/p}/p^6)$



Random Order Median Finding

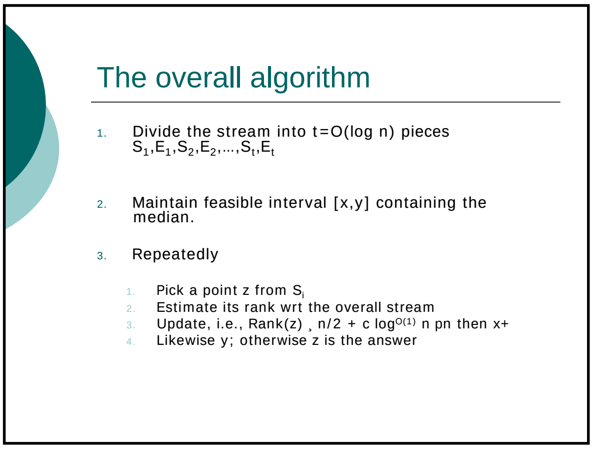
- Guha, McGregor `07
  - O(1) space, § pn log$^{O(1)}$ n approximation

) O(log log n) passes
- $\Omega(n^{1/p}/p^6)$) polylog space implies Ω(log n) passes
- Exponential separation!



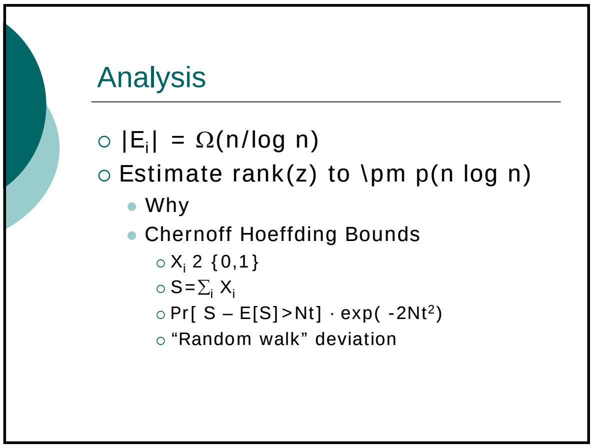
Upper Bound

- Apologies: Its not difficult.
- Sometimes there is only one way of looking at a problem, which makes it obvious in retrospect.
  - Those are the algorithms from the "book".



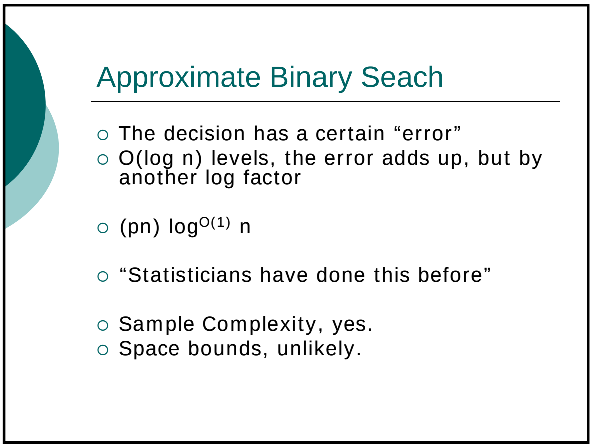
The overall algorithm

1. Divide the stream into t=O(log n) pieces $S_1,E_1,S_2,E_2,\ldots,S_t,E_t$
2. Maintain feasible interval [x,y] containing the median.
3. Repeatedly
   1. Pick a point z from S_i
   2. Estimate its rank wrt the overall stream
   3. Update, i.e., Rank(z) ¸ n/2 + c log$^{O(1)}$ n pn then x+
   4. Likewise y; otherwise z is the answer



Analysis

- $|E_i|$ = Ω(n/log n)
- Estimate rank(z) to \pm p(n log n)
  - Why
  - Chernoff Hoeffding Bounds
    - X_i 2 {0,1}
    - $S=\Sigma_i X_i$
    - Pr[S – E[S]>Nt] · exp(-2Nt^2)
    - "Random walk" deviation



Approximate Binary Seach

- The decision has a certain "error"
- O(log n) levels, the error adds up, but by another log factor
- (pn) log$^{O(1)}$ n
- "Statisticians have done this before"
- Sample Complexity, yes.
- Space bounds, unlikely.

Lower Bounds

- Recall Indexing
  - Alice has σ 2 $\{0,1\}^n$
  - Bob has j
  - Need $\sigma[j]$
- Must send $\Omega(n)$ bits

A reduction of median to Indexing

- Alice creates/adds to the stream
  - $2i+\sigma[i]$
  - Starts running the median finding alg.
  - Sends the state of memory to Bob
- Bob adds
  - n-j copies of (-1)'s
  - J-1 copies of (2n+2)'s
- Median is ?

Approximate medians to Indexing

- $n=1/\varepsilon$
- $\Omega(1/\varepsilon)$ bound
- How to extend to multiple passes?

Round Elimination Lemma

- Bro-Miltersen, Nisan, Safra, Wigderson '98
- Number of bits in a round: S
- f is any communication problem
- Define P_f
  - Alice has $x_1,x_2,\ldots,x_m$
  - Bob has j,y
  - Need $f(x_j,y)$
- A k round S bit protocol for P_f implies a (k-1) round 2S bit protocol for f
- $N^{1/p}/2^p$ lower bound for p-round protocols

Multipass) Multiround

- Alice dumps data to first part
- Bob dumps to second half
- K passes
- $\underbrace{\overbrace{A,B},A,B,\ldots,\ldots,\ldots,\ldots,A,B}_{K}$

2k-1 Rounds

Consequence

- R() is a mapping for f
- Alice creases $R(x_1),R(x_2),\ldots,R(x_m)$
- Bob creases R(y) & the **selector**
- **Median:**
  - (n-j) |R| copies -1
  - (j-1) |R| copies +1

Gap in Exponent

- 1/k versus 1/(2k-1)
- In sublinear space algorithms the **holy grail is the exponent**!
- Two roads …

Road 1: Multiplayer Communication Complexity

- Pointer Chasing,
  - Nisan, Wigderson '93
- K+1 players
- Works when players have "similar" category of input – good for permutation invariant function
- (Median qualifies)
- $N^{1/k}/k^{O(1)}$ bound for k passes

Road 2: Pass Elimination Lemma

- Guha, McGregor 'xx

1. CC) Stream will always have a blowup in factor 2 in passes) rounds
2. Interaction
   - Alice has $x_1,x_2,\dots,x_m$
   - Bob has y
   - Interaction between x_i,x_j for ordered problems

- Prove something directly on streams.

Pass Elimination

- Define P_f for any streaming function f
  - You are given $x_1,x_2,\dots,x_m,i$
  - Compute $f(x_i)$
- This is not a communication problem – 2 rounds!
- S space k pass algorithm for P_f gives a 2S space k-1 pass algorithm for f
- **Note**:
  - Reduction has to be a streaming algorithm
  - Simpler than CC proofs! There is no Bob.

Go forth & prove your lower bounds …

- $N^{1/k}/2^k$ lower bounds for a variety of problems
- Also gives a Direct-sum type byproduct:
  - suppose we wish to solve all $f(x_i)$.
  - Space S alg.) space S/m alg. for f()

That's all folks