Error Correcting Codes: Combinatorics, Algorithms and Applications **(Fall 2007)**

Lecture 14: Randomized Communication Complexity

2-16-09

Lecturer: Atri Rudra *Scribe: Jeff Hazel*

1 Randomized Communication Complexity

Recall:
Alice has x and Bob has y. They both wish to have f(x,y) and seek to minimize the number of bits they must transmit.

Consider:
Would having access to an unlimited number of random bits affect the communication complexity of computing f(x,y)?

2 EX: Given X = [x_1....x_n] and Y = [y_1....y_n], does x_i = $y_i \forall$ i ?

idea:
pick c: $[0,1]^n \rightarrow [0,1]^m$ s.t. dist(c) $\geq \Omega$(m)

protocol:
Alice: choose a random i $\in [m]$
Alice: computes b = $c(x)_i$
Alice: sends $[b, i]$ to Bob
Bob: computes c(y) and checks if b = $c(y)_i$

case 1: c(x) = c(y): $\rightarrow$ Pr$[\tau(x,y) = 0]$ = 0
case 2: c(x) $\neq$ c(y): Δ(c(x),c(y)) $\geq \delta$(m) $\rightarrow$ Pr$[\tau(x,y) = 1] \leq$ 1 - τ
$\rightarrow$ communication: O(log m) bits
$\rightarrow$ NEED: 1 - $\delta <$ 1/3, $\delta \geq$ 2/3
$\rightarrow$ repeat the protocol 0(n)times to decrease prob of bad answer.

We Must Pick a Code . . .

i. Hadamard: C = HAD $\rightarrow$ $CC_{1/3}$(EQ) $\leq$ O(n)

ii. Asymptotically good code: C $\rightarrow$ $CC_{1/3}$(EQ) $\leq$ O(log n)

$$CC_{1/n}(EQ) \leq O(\log^2 n)$$

Q: Can we do better? . . . .

TRICK:
pick c to be a q-ary code s.t.:

$$n^2 \leq q \leq 2n^2$$

CC = O(log m + log q), $\delta \geq 1 - 1/n$

NEED: *1: log m = O(log n), *2: $\delta \geq 1 - 1/n$
REED-SOLOMON: $[m, n, m-n+1]_q$ codes $q \leq m$.

pick q s.t. it is a prime power

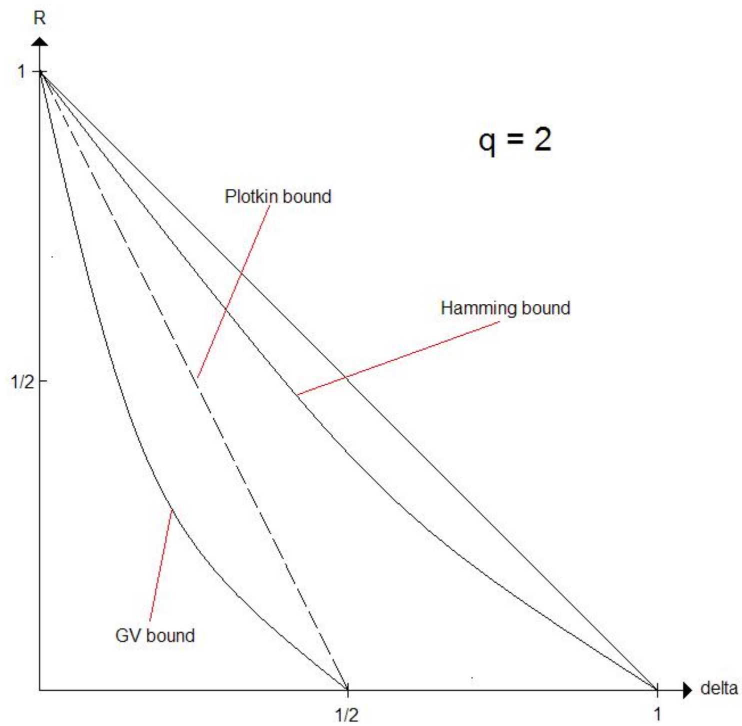
$\downarrow$

power of 2 $\rightarrow n^2 \leq q \leq 2n^2$

for simplicity: m = q

$\rightarrow$ *1: log m = O(log n), since $q \leq 2n^2$
$\rightarrow$ *2: $\delta \geq 1 - n/m \geq 1 - 1/n$ (because $n^2 \leq m$)
$\rightarrow CC_{1/n}(EQ) = O(\log n)$

R vs δ


R
1
1/2
q = 2
Plotkin bound
Hamming bound
GV bound
delta
1/2
1