Communication Complexity

Jie Ren

Adaptive Signal Processing and Information Theory Group

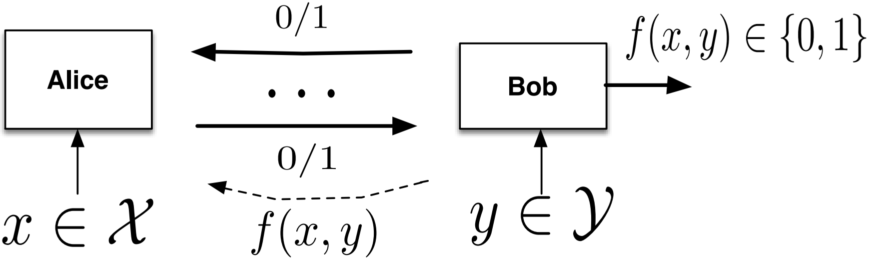
Nov 3^{rd}, 2014

1. E. Kushilevitz and N. Nisan, "Communication Complexity," Cambridge University Press, 1997.
2. L. Lovasz, "Communication Complexity: A Survey," in Paths, Flows, and VLSI Layout, B. H. Korte, Ed., Springer Verlag: Berlin 1990.
3. T. Lee and A. Shraibman, "Lower Bounds in Communication Complexity: A Survey," Now Publishers Inc., 2009.
4. A. C. Yao, "Some Complexity Questions Related to Distributed Computing," Proc. of 11th ACM Symposium on Theory of Computing, 1981, 308-311.
5. P. Beame and J. Lawry, "Randomized versus Nondeterministic Communication Complexity," Proc. of 24th ACM Symposium on Theory of Computing, 1992, 188-199.
6. A. K. Chandra, M. L. Furst and R. J. Lipton, "Multi-party Protocols," Proc. of 15th ACM Symposium on Theory of Computing, 1983, 94-99.
7. P. B. Miltersen, N. Nisan, S. Safra and A. Wigderson, "On Data Structures and Asymmetric Communication Complexity," Proc. of 27th ACM Symposium on Theory of Computing, 1995, 103-111.

Outline

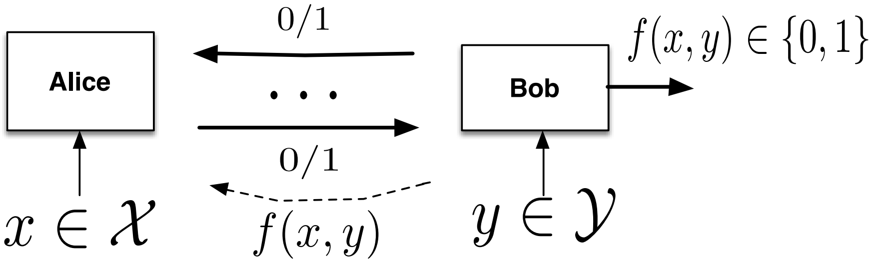
Outline

Problem Setup



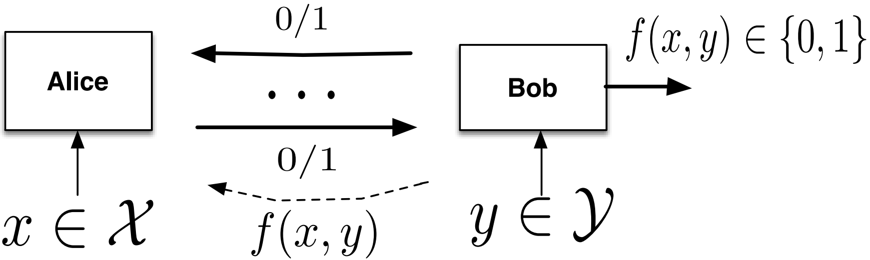
- Two party communication
- Each knows an input $x \in \mathcal{X} / y \in \mathcal{Y}$
- Let one/both sides compute a function f with no error
- Only care about communication cost

Problem Setup



- Sending binary messages
- f usually binary
- Deterministic protocol $\mathcal{P}$: who to talk/what to send
- Communication cost: sum of total bits/rounds $CC(\mathcal{P})$

Deterministic Communication Complexity



$$D(f) = \min_{\mathcal{P}} \max_{(x,y)\in X\times Y} CC(\mathcal{P}) \quad (1)$$

A Naive Upper Bound

Proposition (naive upper bound): For every function $f : X \times Y \to Z$

$$D(f) \leq \log_2 |X| + \log_2 |Z| \tag{2}$$

A Naive Upper Bound

Example: MAX of the union

Alice and Bob hold subsets $x, y \subseteq \{1, \ldots, n\}$ respectively, and they with to compute $MAX(x, y)$.

$$D(MAX) \leq 2\log_2 n \tag{3}$$

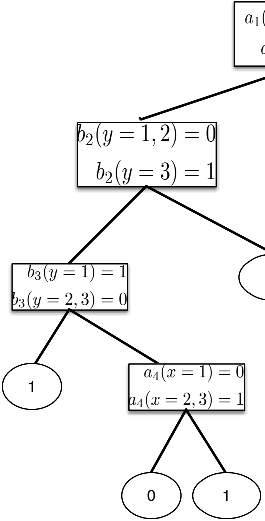
Outline

Definition: Protocol Tree

Definition

: A protocol $\mathcal{P}$ over domain $X \times Y$ with range Z is a binary tree where each internal node v is labeled either by a function $a_v : X \rightarrow \{0, 1\}$ or by a function $b_v : Y \rightarrow \{0, 1\}$, and each leaf is labeled with an element $z \in Z$. The communication cost $CC(\mathcal{P})$ will be the depth of the protocol tree.

Example: Protocol Tree



$$f(x, y) = \begin{cases} 1 & x \geq y \\ 0 & \text{OTH} \end{cases}$$

$$x \in \{1, 2, 3\}$$

$$y \in \{1, 2, 3\}$$

Why Binary Message?

- Entropy not involved - simple?
- No block coding (compute a single function)
- Worst case - always exists $p = 1/2$ s.t. $h_2(p) = 1$

One Side Compute f Vs. Both Sides Compute f

- Equivalent setup
- $\Rightarrow$ Need one more bit if f is binary
- $\Leftarrow$ Second last round: one side must know $f(x, y)$

Some Lower Bounds of Communication Complexity

- $D(f)$: unknown for general f
- Interested in lower bounds

Outline

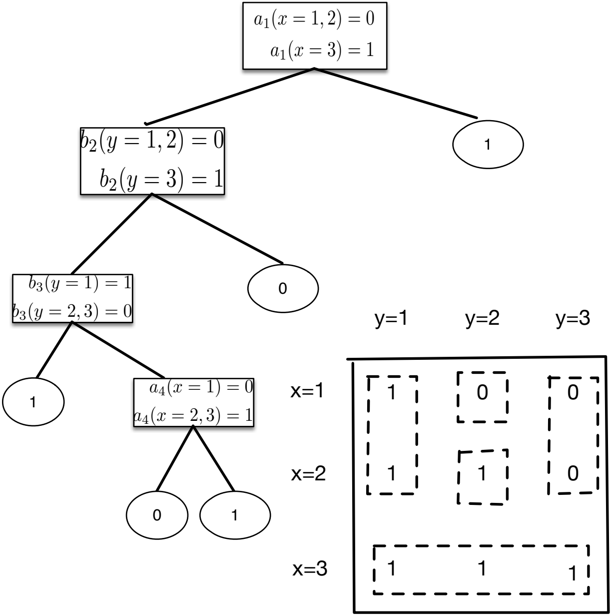
Combinatorial Rectangles

Definition: Let $\mathcal{P}$ be a protocol and v be a node of the protocol tree. R_v is the set of inputs (x, y) that reach node v.
Proposition: If L is the set of leaves of a protocol $\mathcal{P}$, then $\{R_\ell, \ell \in L\}$ is a partition of $X \times Y$.
Definition: A combinatorial rectangle is a subset $R \subseteq X \times Y$ such that $R = A \times B$ for some $A \subseteq X$ and $B \subseteq Y$.

Proposition: $R \subseteq X \times Y$ is a rectangle iff

$$(x_1, y_1) \in R \;\&\; (x_2, y_2) \in R \Rightarrow (x_1, y_2) \in R. \tag{4}$$

Proposition: For every protocol $\mathcal{P}$ and leaf ℓ, R_ℓ is a rectangle



$a_1(x=1,2)=0$
$a_1(x=3)=1$
$b_2(y=1,2)=0$
$b_2(y=3)=1$
1
$b_3(y=1)=1$
$b_3(y=2,3)=0$
0
1
$a_4(x=1)=0$
$a_4(x=2,3)=1$
0
1
y=1
y=2
y=3
x=1
1
0
0
x=2
1
1
0
x=3
1
1
1

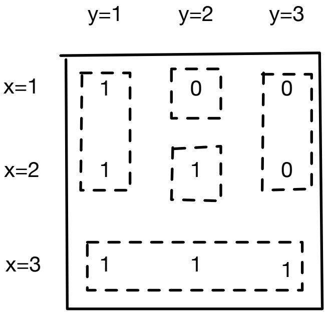
Rectangle lower bound

Definition: A subset $R \subseteq X \times Y$ is called f-monochromatic if f is fixed on R.

Theorem 1.17 (Kushilevitz & Nisan): If any partition of $X \times Y$ into f-monochromatic rectangles requires at least t rectangles, then

$$\log_2 t \leq D(f) \tag{5}$$

- $\mathcal{P}$ partitions $X \times Y$ into monochromatic rectangles
- Depth of its protocol tree: $\geq \log_2 t$

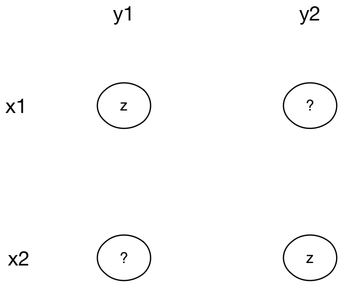


$$D(f) \geq \log_2 5$$

$$D(f) = 3$$

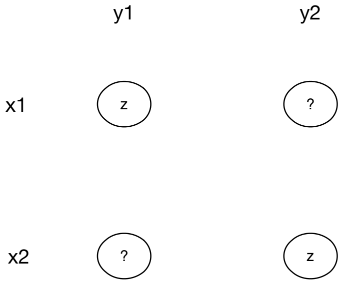
Outline

Motivation: If we exhibit a large set of input pairs such that no two of them can be in a single monochromatic rectangle, then the number of partitions of $\mathcal{P}$ must be large



Definition : Let $f : X \times Y \rightarrow \{0,1\}$. A set $S \subseteq X \times Y$ is called a fooling set if there exits a value $z \in \{0,1\}$ such that

- For every $(x,y) \in S$, $f(x,y) = z$
- For every two distinct pairs (x_1, y_1) and (x_2, y_2) in S, either $f(x_1, y_2) \neq z$ or $f(x_2, y_1) \neq z$



Fooling set lower bound

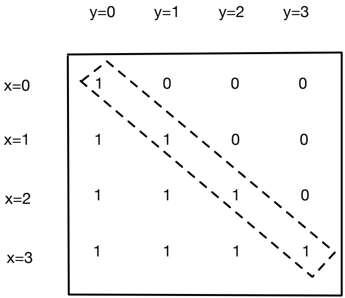
Theorem 1.20 (Kushilevitz & Nisan) : If f has a fooling set S of size t, then

$$\log_2 t \leq D(f) \tag{6}$$

Proof : No monochromatic rectangle contains more than one element of S

Example: Alice and Bob each hold an n-bit integer $0 \leq x, y < 2^n$. The "greater than or equal to" function, $GTE(x, y)$, is defined to be 1 iff $x \geq y$.

$$D(GT) = n + 1 \tag{7}$$



Example: Alice and Bob each hold an n-bit integer $0 \leq x, y < 2^n$. The "greater than or equal to" function, $GTE(x, y)$, is defined to be 1 iff $x \geq y$.

$$D(GT) = n + 1 \tag{8}$$

	y=0	y=1	y=2	y=3
x=0	1	0	0	0
x=1	1	1	0	0
x=2	1	1	1	0
x=3	1	1	1	1

Outline

Motivation: Give communication complexity lower bound in an algebraic way

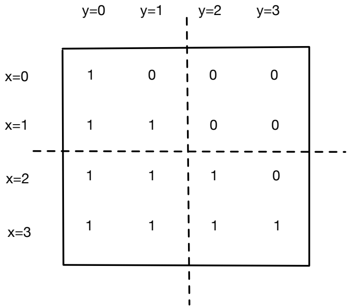
Definition: Associate with every function $f : X \times Y \to \{0,1\}$ a matrix M_f of dimensions $|X| \times |Y|$. The rows/columns of M_f are indexed by the elements of X/Y. Then $rank(f)$ is the linear rank of M_f over the field of reals.

	y=0	y=1	y=2	y=3
x=0	1	0	0	0
x=1	1	1	0	0
x=2	1	1	1	0
x=3	1	1	1	1

Rank lower bound

Theorem 1.28 (Kushilevitz & Nisan): For any function $f : X \times Y \to \{0,1\}$

$$\log_2 rank(f) \le D(f) \tag{9}$$



Proof: Let L_1 be the set of leaves in which the output is 1. For each $\ell \in L_1$,

$$M_\ell(x,y) = \begin{cases} 1 & \text{if } (x,y) \in R_\ell \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$M_f = \sum_{\ell \in L_1} M_\ell \tag{11}$$

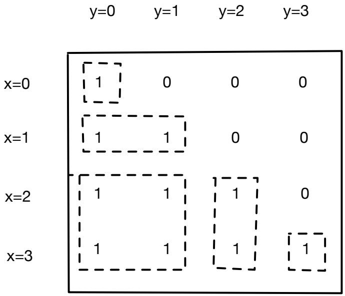
and

$$rank(M_f) \le \sum_{\ell \in L_1} rank(M_\ell) \le |L_1| \le |L| \tag{12}$$

Rank lower bound

Theorem 1.28 (Kushilevitz & Nisan): For any function $f : X \times Y \to \{0, 1\}$

$$\log_2 rank(f) \leq D(f) \tag{13}$$



Rank upper bound

Proposition 2.3 (Lovasz 1990): For any function $f : X \times Y \to \{0,1\}$

$$D(f) \leq rank(f) \tag{14}$$

Proof: We know that row rank = column rank = rank(f), and we can form the row vector space with dimension rank(f). We then claim that there are at most $2^{rank(f)}$ distinct row vectors, the reason is because, although the coefficients for the polynomials that represent the row vectors can be real, the entries of the matrix $M(f)$ can only be 0 or 1. Hence we can build a protocol as follows: Alice merge the repeated rows of $M(f)$ on the table to have $M'(f)$, and then sends the index of row in $M'(f)$ that contains x. Bob compute $f(x,y)$ based on what he received.

Summary

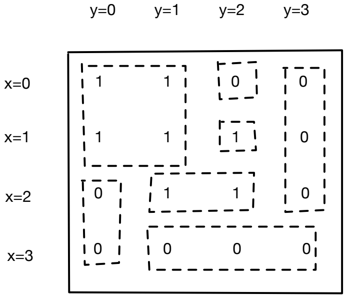
- Concept: protocol tree, combinatorial rectangles, fooling sets, rank
- Naive upper bound: $\log_2 |X| + 1$
- Rectangle lower bound: $\log_2 t_r$
- Fooling set lower bound: $\log_2 t_f$
- Rank lower bound: $\log_2 rank(f)$
- Rank upper bound: $rank(f)$

Outline

Outline

Motivation

- How good are the rectangle lower bounds?
- Relaxing the need to partition by allowing covering of the same space



Motivation: Alice has a n-bit string $x \in \{0,1\}^n$, Bob has a n-bit string $y \in \{0,1\}^n$, either side wants $NEQ(x,y)$.

$$D(NEQ) = n \tag{15}$$

Now assume a third person knows everything: x,y and $NEQ(x,y)$ and want to convince Alice and Bob, Alice and Bob need to check the correctness

- Sends the index of the first bit differs

Setup: A prover, who sees both x and y, is trying to convince Alice and Bob that "$f(x, y) = 1$". If $f(x, y) \neq 1$, then Alice and Bob must be able to detect the proof is wrong.

$$f(x, y) = 1 \Rightarrow \exists\ z\ \mathcal{P}(x, y, z) = 1 \tag{16}$$

$$f(x, y) = 0 \Rightarrow \forall\ z\ \mathcal{P}(x, y, z) = 0 \tag{17}$$

- Two-stage nondeterministic protocol $\mathcal{P}^N$
  1. Alice and Bob receive a message z from the third person.
  2. Alice and Bob run a deterministic protocol $\mathcal{P}^{D,z}$ based on z.
- The interesting cost in this protocol is the maximum length of z plus the number of bits exchanged over all x, y.

Alternative Setup: Let $f : X \times Y \to \{0,1\}$ be a function. Let $L = \{(x,y) : f(x,y) = 1\}$. A successful nondeterministic protocol for f consists of functions $f_A : X \times \{0,1\}^k \to \{0,1\}$ and $f_B : Y \times \{0,1\}^k \to \{0,1\}$ such that

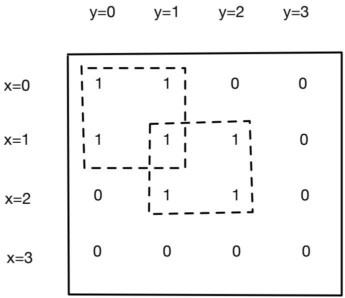
1. $\forall (x,y) \in L$, $\exists z \in \{0,1\}^k$ s.t. $f_A(x,z) \wedge f_B(y,z) = 1$
2. $\forall (x,y) \notin L$, $\forall z \in \{0,1\}^k$, $f_A(x,z) \wedge f_B(y,z) = 0$

- One stage nondeterministic protocol
  1. Alice and Bob receive a message z and compute $f(x,y)$ successfully.
- The interesting cost in this protocol is the length of z only.

Two-stage $\Rightarrow$ One-stage: Given a two-stage nondeterministic protocol with k bits first stage cost and d bits second stage cost, we can always build a one-stage nondeterministic protocol by adding the d bits deterministic communication to the witness z with each party accepting if the message agrees with what Alice and Bob would have said in the protocol.

In the language of "Rectangles": A prover, who sees both x and y, is trying to convince Alice and Bob that "$f(x, y) = 1$" by broadcasting a 1-monochromatic rectangle that **cover** (x, y).

- By "Nondeterministic" we mean: this 1-monochromatic rectangle may not be unique



Outline

Definitions: Let $f : X \times Y \to \{0,1\}$ be a binary function.

- $C^{\mathcal{P}}(f)$: the smallest number of leaves in a protocol $\mathcal{P}$
- $C^{D}(f)$: the smallest number of monochromatic rectangles that partition $X \times Y$
- $C(f)$: the smallest number of monochromatic rectangles needed to cover $X \times Y$
- $C^{z}(f)$: the smallest number of monochromatic rectangles needed to cover the z-inputs

Proposition 2.2 (Kushilevitz & Nisan): For all $f : X \times Y \to \{0,1\}$,

$$\log_2\left(C^0(f) + C^1(f)\right) \le \log_2 C^D(f) \le \log_2 C^{\mathcal{P}}(f) \le D(f) \tag{18}$$

Theorem 29 (Lee & Shraibman): Let $f : X \times Y \to \{0,1\}$ be a function,

$$N^1(f) = \lceil \log_2 C^1(f) \rceil \tag{19}$$

Proof:

- $N^1(f) \le \lceil \log_2 C^1(f) \rceil$: Let $\{R_\ell\}$ be a cover. If $f(x,y) = 1$, the players receive the index ℓ that $(x,y) \in R_\ell$
- $N^1(f) \ge \lceil \log_2 C^1(f) \rceil$: Let $k = N^1(f)$, and let $f_A : X \times \{0,1\}^k \to \{0,1\}$, $f_B : Y \times \{0,1\}^k \to \{0,1\}$ be functions in the one-stage nondeterministic protocol. Define $R_z = \{(x,y) : f_A(x,z) \wedge f_B(y,z) = 1\}$, R_z is a rectangle. We claim $\{R_z, z \in \{0,1\}^k\}$ is a cover of the 1s. This is because by the definition of nondeterministic protocol:
    - $\forall (x,y)$ pairs that $f(x,y) = 1$, there must exists some z s.t. $(x,y) \in R_z$.
    - $\forall (x,y)$ pairs that $f(x,y) = 0$, $(x,y) \notin R_z$ for all z.

Definition (Lee & Shraibman):

$$N^1(f) = \lceil \log_2 C^1(f) \rceil \tag{20}$$

$$N^0(f) = \lceil \log_2 C^0(f) \rceil \tag{21}$$

$$N(f) = \max(N^1(f), N^0(f)) \tag{22}$$

Definition (Kushilevitz & Nisan):

$$N^1(f) = \log_2 C^1(f) \tag{23}$$

$$N^0(f) = \log_2 C^0(f) \tag{24}$$

$$N(f) = \log_2 \left(C^0(f) + C^1(f)\right) \tag{25}$$

Outline

Motivation:

- Introduce randomness in the protocol r_A and r_B: flip coins
- Allow protocols that may have error

Randomized Protocol Tree

Definition

: A randomized protocol $\mathcal{P}$ over domain $X \times Y$ with range Z is a binary tree where each internal node v is labeled either by a function $a_v : X \times R_A \to \{0, 1\}$ or by a function $b_v : Y \times R_B \to \{0, 1\}$, and each leaf is labeled with an element $z \in Z$.

Definition: Let $\mathcal{P}$ be a randomized protocol. All the probabilities below are over random choices of r_A and r_B.

- $\mathcal{P}$ computes a function f with zero error
- $\mathcal{P}$ computes a function f with $\epsilon-$error if for all (x, y)

$$\mathbb{P}[\mathcal{P}(x, y) = f(x, y)] \geq 1 - \epsilon \tag{26}$$

- $\mathcal{P}$ computes a function f with one-sided $\epsilon-$error if for all (x, y) such that $f(x, y) = 0$

$$\mathbb{P}[\mathcal{P}(x, y) = 0] = 1, \tag{27}$$

  and for all (x, y) such that $f(x, y) = 1$,

$$\mathbb{P}[\mathcal{P}(x, y) = 1] \geq 1 - \epsilon \tag{28}$$

Definition: Let $f : X \times Y \to \{0, 1\}$ be a binary function. We consider the following complexity measure for f

- $R_0(f)$ is the minimum average case cost of a randomized protocol that computes f with zero error
- $R_\epsilon(f)$ is the minimum worst case cost of a randomized protocol that computes f with error ϵ. We typically use $\epsilon = 1/3$
- $R^1_\epsilon(f)$ is the minimum worst case cost of a randomized protocol that computes f with one-sided error ϵ.

Why we care these measures:

- worst case zero error = deterministic
- for all average case ϵ error, there exists a worst case problem that can convert to

Proposition: Given a protocol $\mathcal{P}$ that makes an error $\epsilon/2$ and the average number of bits exchanged is t, it can always be modified as follows: execute $\mathcal{P}$ as long as at most $2t/\epsilon$ bits are exchanged, if the protocol finishes, use its output, otherwise output 0. This gives a worst case cost $2t/\epsilon$ with error upper bounded by ϵ.

Proof:

$$\begin{aligned} t &= \sum_{r_a,r_b,x,y} cc \cdot p(r_a, r_b, x, y) \\ &= \sum_{cc \le 2t/\epsilon} cc \cdot p(r_a, r_b, x, y) + \sum_{cc > 2t/\epsilon} cc \cdot p(r_a, r_b, x, y) \\ &\ge \sum_{cc > 2t/\epsilon} cc \cdot p(r_a, r_b, x, y) \ge \frac{2t}{\epsilon} Pr[cc > 2t/\epsilon] \end{aligned} \tag{29}$$

Hence,

$$\begin{aligned} Pr[err] &\le \frac{\epsilon}{2} Pr[\mathcal{P} \ ends] + 1 \cdot Pr[\mathcal{P} \ not \ ends] \\ &\le \frac{\epsilon}{2} + \frac{t}{2t/\epsilon} = \epsilon \end{aligned} \tag{30}$$

1. For all $0 < \epsilon \leq \epsilon^{'} < 1/2$,

$$R_\epsilon(f) \leq O(\log_{\epsilon'} \epsilon \cdot R_{\epsilon'}(f)) \tag{31}$$

2. For all $0 < \epsilon \leq 1/2$,

$$R_\epsilon(f) \leq R^1_\epsilon(f) \leq O(\log \epsilon^{-1} R_0(f)) \tag{32}$$

3. For all $0 < \epsilon \leq 1/2$,

$$R_0(f) = \Theta(\max[R^1_\epsilon(f), R^1_\epsilon(not(f))]) \tag{33}$$

Proof of Property 1: We first prove a similar result for the 1-sided error problem: for all $0 < \epsilon \leq \epsilon^{'} < 1/2$,

$$R_{\epsilon}^{1}(f) \leq O(\log_{\epsilon^{'}} \epsilon \cdot R_{\epsilon^{'}}^{1}(f)) \tag{34}$$

Proof of Property 1: Given a randomized protocol $\mathcal{P}$ with worst case cost T bits and one-sided error no greater than $\epsilon^{'} < 1/2$, we can build a new protocol $\mathcal{P}^{'}$ with worst case cost nT bits by simply running $\mathcal{P}$ n times. In the new protocol, Alice and Bob will claim $f(x, y) = 1$ if and only if there exists at least one time among the n repeating protocols that they will output 1. Now we bound the error for the new protocol $\mathcal{P}^{'}$:

$$\mathbb{P}[err|f(x,y) = 0] = 0 \tag{35}$$

$$\begin{aligned} \mathbb{P}[err|f(x,y) = 1] &= \mathbb{P}[\text{all n trails output } 0|f(x,y) = 1] \\ &= (\epsilon^{'})^n \end{aligned} \tag{36}$$

Therefore, if we repeat $\mathcal{P}$ $\log_{\epsilon^{'}} \epsilon$ times, we can guarantee to reduce the one-sided error to ϵ.

Proof of Property 1: Now we prove property 1. We still run $\mathcal{P}$ n times, each gives an output $X_i, i \in \{1, \dots, n\}$. In the new protocol, Alice and Bob will claim $f(x, y) = 1$ if and only if

$$\frac{1}{n}\sum_i X_i > \frac{1}{2} \tag{37}$$

Now we bound the error for the new protocol $\mathcal{P}^{'}$:

$$\begin{aligned}\mathbb{P}[err|f(x,y)=0] &\leq \sum_{i=\lceil n/2 \rceil}^{n} \binom{n}{i} (\epsilon^{'})^i (1-\epsilon^{'})^{n-i} \\ &\leq (\epsilon^{'})^n\end{aligned} \tag{38}$$

$$\begin{aligned}\mathbb{P}[err|f(x,y)=1] &\leq \sum_{i=\lceil n/2 \rceil}^{n} \binom{n}{i} (\epsilon^{'})^i (1-\epsilon^{'})^{n-i} \\ &\leq (\epsilon^{'})^n\end{aligned} \tag{39}$$

Therefore, if we repeat $\mathcal{P}$ $\log_{\epsilon^{'}} \epsilon$ times, we can also guarantee to reduce the two-sided error to ϵ.

Outline

Distributional Complexity

Motivation: Consider probability distributions over the inputs
Definition: Let μ be a probability distribution on $X \times Y$. The (μ, ϵ)-distributional communication complexity of f, $D^{\mu}_{\epsilon}(f)$, is the cost of the best deterministic protocol that gives the correct answer for f with a probability at least $1 - \epsilon$.

Discrepancy

Motivation: Allow those rectangles that partition the support to be "almost" f-monochromatic.

Definition: Let $f : X \times Y \rightarrow \{0,1\}$ be a function, R be any rectangle, and μ be a probability distribution on $X \times Y$, Denote

$$Disc_{\mu}(R, f) = |\underset{\mu}{\mathbb{P}}[f(x,y) = 0 \text{ \& } (x,y) \in R] - \underset{\mu}{\mathbb{P}}[f(x,y) = 1 \text{ \& } (x,y) \in R]| \tag{40}$$

The discrepancy of f according to μ is,

$$Disc_{\mu}(f) = \max_{R} Disc_{\mu}(R, f) \tag{41}$$

Discrepancy

Proposition 3.28 (Kushilevitz & Nisan): For every function $f : X \times Y \to \{0, 1\}$, every probability distribution μ on $X \times Y$, and every $\epsilon \geq 0$,

$$D^{\mu}_{1/2-\epsilon}(f) \geq \log_2(2\epsilon/Disc_{\mu}(f)) \tag{42}$$

Discrepancy

Proof: Given any $\mathcal{P}$ with c bits to compute f, we have

$$
\begin{aligned}
2\epsilon &\leq \mathbb{P}[\mathcal{P}(x,y) = f(x,y)] - \mathbb{P}[\mathcal{P}(x,y) \neq f(x,y)] \\
&= \sum_{\ell} (\mathbb{P}[\mathcal{P}(x,y) = f(x,y) \& (x,y) \in R_\ell] \\
&-\mathbb{P}[\mathcal{P}(x,y) \neq f(x,y) \& (x,y) \in R_\ell]) \\
&\leq \sum_{\ell} |\mathbb{P}_{\mu}[f(x,y) = 0 \ \& \ (x,y) \in R_\ell] - \mathbb{P}_{\mu}[f(x,y) = 1 \ \& \ (x,y) \in R_\ell]| \\
&\leq 2^c \cdot Disc_\mu(f)
\end{aligned} \tag{43}
$$

Outline

Recall

Definitions: Let $f : X \times Y \to \{0, 1\}$ be a binary function.

- $C^{\mathcal{P}}(f)$: the smallest number of leaves in a protocol $\mathcal{P}$
- $C^{D}(f)$: the smallest number of monochromatic rectangles that partition $X \times Y$
- $C(f)$: the smallest number of monochromatic rectangles needed to cover $X \times Y$
- $C^{z}(f)$: the smallest number of monochromatic rectangles needed to cover the z-inputs

Recall

Proposition: For all $f : X \times Y \to \{0,1\}$,

$$\log_2 C(f) \leq \log_2 C^D(f) \leq \log_2 C^{\mathcal{P}}(f) \leq D(f) \tag{44}$$

$$C(f) = C^0(f) + C^1(f) \tag{45}$$

Definition: The nondeterministic communication complexity,

$$N^1(f) = \log_2 C^1(f) \tag{46}$$

$$N^0(f) = \log_2 C^0(f) \tag{47}$$

$$N(f) = \log_2 C(f) \tag{48}$$

Protocol partition number

Theorem 2.8 (Kushilevitz and Nisan): The protocol partition number of a function determines the deterministic communication complexity.

$$\log_2 C^{\mathcal{P}}(f) \leq D(f) \leq 2\log_{3/2} C^{\mathcal{P}}(f) \tag{49}$$

Proof: Given any protocol $\mathcal{P}$ with t number of leaves, it can be converted into a "balanced" protocol.

Protocol partition number

Proof: Given any protocol $\mathcal{P}$ with t number of leaves, there must exist an internal node v such that

$$t/3 < t_v \leq 2t/3 \tag{50}$$

We build a new protocol based on this internal node:

1. Alice and Bob determine whether or not $(x, y) \in R_v$
2. If $(x, y) \in R_v$, Alice and Bob recursively solve f in the rectangle R_v.
3. If $(x, y) \notin R_v$, Alice and Bob recursively solve f' on $X \times Y$ where

$$f'(x_1, y_1) = \begin{cases} f(x_1, y_1) & \text{if } (x_1, y_1) \notin R_v \\ 0 & \text{otherwise} \end{cases} \tag{51}$$

Protocol partition number

Analysis:

- Step 1 Requires 2 bits
- In Step 3, we take $\mathcal{P}$ and replace $Tree(v)$ by a 0-leaf, we get a protocol for f' with $t - t_v + 1$ leaves, hence

$$D(t) \leq 2 + D(2t/3) \tag{52}$$

Recall

Proposition: For all $f : X \times Y \to \{0,1\}$,

$$\log_2 C(f) \leq \log_2 C^D(f) \leq \log_2 C^{\mathcal{P}}(f) \leq D(f) \leq 2\log_{3/2} C^{\mathcal{P}}(f) \tag{53}$$

Definition: The nondeterministic communication complexity,

$$N^1(f) = \log_2 C^1(f) \tag{54}$$

$$N^0(f) = \log_2 C^0(f) \tag{55}$$

$$N(f) = \log_2 C(f) \tag{56}$$

Deterministic CC Vs. Nondeterministic CC

How good is the rectangle lower bound?

$$D(f) \stackrel{?}{=} O(\log C^D(f)) \tag{57}$$

Deterministic CC Vs. Nondeterministic CC

Theorem 2.11 (Kushilevitz & Nisan): For every function $f : X \times Y \to \{0, 1\}$,

$$D(f) = O(N^0(f)N^1(f)) \tag{58}$$

Deterministic CC Vs. Nondeterministic CC

Property: Let $R = S \times T$ be a 0-monochromatic rectangle, and let $R' = S' \times T'$ be a 1-monochromatic rectangle, then either $S \cap S' = \emptyset$ or $T \cap T' = \emptyset$.

Deterministic CC Vs. Nondeterministic CC

Proof of Theorem 2.11 (Kushilevitz & Nisan): We give a protocol $\mathcal{P}$ as follows, Alice and Bob search for a 0-rectangle that contains the input (x, y), and they conclude $f(x, y) = 1$ if they fail. In each round, Alice and Bob exchange $\log_2 C^1(f) + 1$ bits and reduce the number of "alive" 0-rectangles by a factor of 2. There will be no more than $\log_2 C^0(f)$ rounds, hence

$$D(f) \leq CC(\mathcal{P}) = O(\log_2 C^0(f)(\log_2 C^1(f) + 1)) = O(N^0(f)N^1(f)) \quad (59)$$

Deterministic CC Vs. Nondeterministic CC

Proof of Theorem 2.11 (Kushilevitz & Nisan): In each round, the players do the following:

1. Alice outputs $f(x, y) = 0$ if no 0-rectangles are alive. Otherwise, Alice looks for a 1-rectangle that contains row x and intersects in rows with at most half of the alive 0-rectangles and send the name of this 1-rectangle.
2. Bob looks for a 1-rectangle that contains column y and intersects in columns with at most half of the alive 0-rectangles and send the name of this 1-rectangle. Otherwise, Bob outputs $f(x, y) = 0$

Deterministic CC Vs. Nondeterministic CC

Protocol Analysis:

- If $f(x, y) = 0$, it must belong to some 0-rectangle R, then R remains alive during the protocol. Therefore, if no 0-rectangle is alive, $f(x, y)$ must be 1
- If neither Alice nor Bob can find a 1-rectangle to announce (which means both of them output $f(x, y) = 1$), we claim this output must be correct by the property.

Public coin

Theorem (Theo. 3 in Newman 1991, Theo. 3.14 in K & N): Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a function. For every $\delta > 0$ and every $\epsilon > 0$, we have

$$R_{\delta+\epsilon}(f) \leq R_{\epsilon}^{pub}(f) + O(logn + log\delta^{-1}) \tag{60}$$

- $\exists$ a set of $t(\delta, n) = O(n/\delta^2)$ public coin protocols with error $\epsilon + \delta$
- Alice tells Bob which protocol to use $\log_2 t = O(\log n + \log \delta^{-1})$

Randomized CC Vs. Distributional CC

Theorem (Theo. 3 in Yao 1979, Theo. 3.20 in K & N):

$$R_{\epsilon}^{pub}(f) = \max_{\mu} D_{\epsilon}^{\mu}(f) \tag{61}$$

- $\Rightarrow$ The randomized protocol is correct for every distribution μ with probability $\geq 1 - \epsilon$
- $\Leftarrow$ Use min-max theorem of zero-sum game