Chapter 1

Deterministic communication complexity

In communication complexity, two players named *Alice* and *Bob* each receive some input that the other player cannot see and seek to compute some function on their joint input. We wish to determine the minimum number of bits that they must exchange to compute this function. As a result, the main object we will study will not be algorithms, but rather *communication protocols*.

Definition 1 (Protocol)**.** A *(communication) protocol* π is a rooted binary tree $T(\pi)$ with the following additional information:

- Every internal node of the tree is labelled with A or B, determining whether Alice or Bob sends the next bit.
- Every internal node v labelled with A also has a corresponding function $h_v : \mathcal{X} \to \{0,1\}$ which determines the next bit that Alice sends.
- Similarly, every internal node v labelled with B also has a corresponding function $h_v : \mathcal{Y} \to \{0,1\}$ which determines the next bit that Bob sends.
- The two edges leaving an internal node v are labelled with 0 and 1, respectively.
- Each leaf node is labelled with 0 or 1.

Definition 2 (Function computed by a protocol)**.** A protocol π *computes* the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ if for every input $(x,y) \in \mathcal{X} \times \mathcal{Y}$, the path in π obtained by following the edge labelled with $h_v(x)$ at each internal node labelled by A and $h_v(y)$ at each internal node labelled by B leads to a leaf with the label $f(x,y)$.

Following the standard computer science approach, we measure the communication cost of protocols in the worst-case sense.

Definition 3 (Communication cost)**.** The *communication cost* of a protocol π is

$$\mathrm{cost}(\pi) = \mathrm{height}(T(\pi)),$$

the height of the tree for π or, in other words, the length of the longest path between the root of $T(\pi)$ and any of the leaves in the tree. It corresponds to the maximum number of bits that Alice and Bob exchange when executing the protocol over any of their inputs.

Definition 4 (Deterministic communication complexity)**.** The *deterministic communication complexity* of the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is

$$D(f) = \min_{\pi \text{ computes } f} \mathrm{cost}(\pi)$$

is the minimum communication cost of a protocol that computes f.

In the rest of this chapter, we will aim to determine the deterministic communication complexity for some fundamental functions.

Exercise 1. Do there exist functions $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with communication complexity $D(f) = 0$?

1.1 Equality

We begin by considering the simplest (but also perhaps the most important) function: equality. The function $\textsc{Eq} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is defined by

$$\textsc{Eq}(x,y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases}$$

so that the value of the function is 1 if and only if Alice and Bob's inputs are identical.

Theorem 1. *$D(\textsc{Eq}) \leq n+1$.*

Proof. There is a very simple protocol π with cost $n+1$ that computes the function $\textsc{Eq}$: Alice sends the n bits of her input to Bob, then Bob sends the value 1 if and only if his input y is equal to the input x that he has received. The value of each leaf is set to be the same as the value of the edge that leads to it. □

1.2 General upper bound

The trivial protocol we used to bound the deterministic communication complexity of the $\textsc{Equality}$ function can also be modified slightly to establish a much more general result.

Theorem 2. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$,*

$$D(f) \leq \min\{\lceil \log_2 |\mathcal{X}| \rceil, \lceil \log_2 |\mathcal{Y}| \rceil\} + 1.$$

Proof. We can encode each element in $\mathcal{X}$ with $\lceil \log_2 |\mathcal{X}| \rceil$ bits. We can use such an encoding to design a protocol π with cost $\lceil \log_2 |\mathcal{X}| \rceil + 1$: Alice sends $\lceil \log_2 |\mathcal{X}| \rceil$ bits to communicate the encoding of her input x to Bob, and Bob replies with the value $f(x,y)$. (Note that Bob now knows both x and y so that he can compute this value.)

Similarly, we can design a protocol π' with cost $\lceil \log_2 |\mathcal{Y}| \rceil$ by taking an encoding of the elements in $\mathcal{Y}$ and having Bob send the encoding of his input y to Alice and then having Alice respond with the value $f(x,y)$. (Now Alice is the one who knows both x and y.)

Combining the two protocols above, we obtain that

$$D(f) \leq \min\{\mathrm{cost}(\pi), \mathrm{cost}(\pi')\} = \min\{\lceil \log_2 |\mathcal{X}| \rceil + 1, \lceil \log_2 |\mathcal{Y}| \rceil + 1\},$$

as we wanted to show. □

1.3 Parity

The general upper bound we established in the last section is far from tight in some cases. Consider for example the $\textsc{Parity} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ function defined by

$$\textsc{Parity}(x,y) = \bigoplus_{i=1}^{n} (x_i \oplus y_i)$$

where $\bigoplus_{i=1}^n z_i = \sum_{i=1}^n z_i \pmod 2$ is the parity of the sum and, similarly, $a \oplus b$ is defined to be $a + b \pmod 2$.

Theorem 3. $D(\textsc{Parity}) \leq 2$.

Proof. Consider the protocol π where Alice sends the value $\bigoplus_{i=1}^n x_i$ to Bob, Bob sends the value $\bigoplus_{i=1}^n y_i$, and the leaf at the end of the path labelled with a and b takes the value $a \oplus b$. This protocol has cost 2, since Alice and Bob each send a single bit. And we can verify that π correctly computes the Parity function since on any input $(x, y) \in \{0,1\}^n \times \{0,1\}^n$, the protocol leads to the leaf labelled with

$$\left(\bigoplus_{i=1}^n x_i\right) \oplus \left(\bigoplus_{i=1}^n y_i\right) = \bigoplus_{i=1}^n (x_i \oplus y_i) \ = \textsc{Parity}(x, y). \qquad \square$$

1.4 Median

Define $[n] = \{1, 2, \ldots, n\}$ and $2^{[n]}$ to be the set of subsets of $[n]$. The $\textsc{Median} : 2^{[n]} \times 2^{[n]} \to [n]$ function is defined by

$$\textsc{Median}(S, T) = \text{median}(S \cup T);$$

it is the median element of the multiset obtained by taking the union of Alice and Bob's sets.

Theorem 4. $D(\textsc{Median}) = O(\log^2 n)$.

Proof. Here is a simple protocol that computes the Median function. Alice and Bob start by exchanging the total number of elements in S and in T. Define $k = \lfloor \frac{|S|+|T|}{2} \rfloor$. Alice and Bob now need to identify the k-th smallest element in the multiset $S \cup T$. They can do so using a binary search approach. In the first step of this search, Alice and Bob exchange the number of elements in $S \cap [n/2]$ and $T \cap [n/2]$, respectively. If that number m is $m \geq k$, then we continue the binary search in the range $\{1, 2, \ldots, n/2\}$. Otherwise, we update $k' = k - m$ and search for the k'-th smallest element in $(S \cup T) \cap \{n/2 + 1, \ldots, n\}$.

This binary search process will end at the median element. The search itself requires $O(\log n)$ rounds, and at each round Alice and Bob exchange a number in the range $1, 2, \ldots, n$, which they can do with $O(\log n)$ bits of communication. $\square$

Exercise 2. Prove that $D(\textsc{Min}) = O(\log n)$ and $D(\textsc{Max}) = O(\log n)$.

Exercise 3. Consider the Median' function where we take the median of the simple set (i.e., deleting duplicates) obtained by taking the union $S \cup T$. Prove that $D(\textsc{Median'}) = O(\log^2 n)$.

Exercise 4. Prove that $D(\textsc{Median}) = O(\log n)$.

1.5 Rectangles and partition number

As we have seen in the last sections, we can give upper bounds on the deterministic communication complexity of a given function by designing a communication protocol that computes the function. Our main goal, however, will generally be to prove lower bounds on the communication complexity of various functions. We can do so by analyzing *combinatorial rectangles*.

Definition 5 (Rectangle)**.** A *(combinatorial) rectangle* over the finite set $\mathcal{X} \times \mathcal{Y}$ is a set $S \subseteq \mathcal{X} \times \mathcal{Y}$ defined by $S = A \times B$ for some sets $A \subseteq \mathcal{X}$ and $B \subseteq \mathcal{Y}$.

Definition 6 (f-monochromatism)**.** Given a function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$, a rectangle $R \subseteq \mathcal{X} \times \mathcal{Y}$ is *f-monochromatic* if $f(x, y)$ takes the same value for all $(x, y) \in R$.

Definition 7 (χ)**.** The *partition number* of a function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$, denoted

$$\chi(f),$$

is the minimum number of f-monochromatic rectangles required to partition $\mathcal{X} \times \mathcal{Y}$.

1.6 Partition bound

Lemma 1. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$,*

$$D(f) \geq \log_2 \chi(f).$$

Proof. Let π be any protocol that computes f and has cost $\mathrm{cost}(\pi) = D(f)$. Each leaf in the tree for π corresponds to an f-monochromatic rectangle and the rectangles corresponding to each leaf in the tree partition $\mathcal{X} \times \mathcal{Y}$. Since a binary tree of depth d can have at most 2^d leaves, this means that $\chi(f) \leq 2^{\mathrm{cost}(\pi)} = 2^{D(f)}$. □

Exercise 5. Show that $D(f) \leq (\log_2 \chi(f) + 1)^2$.

Open Problem 1. Find a function f for which $D(f) \geq 4 \log_2 \chi(f)$.

Chapter 2

More lower bound techniques

In the first chapter, we introduced the deterministic communication complexity model and saw the general *partition bound* method for proving communication complexity lower bounds. While the partition bound is quite strong, it is hard to work with directly. The goal of this chapter is to introduce other general methods for proving communication complexity lower bounds more easily: the *fooling set bound*, the *rectangle size bound*, and the *log rank bound*.

2.1 Fooling set bound

For many functions, the *fooling set* bound is the easiest way to get meaningful communication complexity lower bounds.

Definition 8 (Fooling set)**.** A set $F \subseteq \mathcal{X} \times \mathcal{Y}$ is a *fooling set* for the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ if there is a value $b \in \{0,1\}$ such that

1. For every $(x,y) \in F$, $f(x,y) = b$; and

2. For every $(x,y) \neq (x',y') \in F$, we have $f(x,y') \neq b$ or $f(x',y) \neq b$ (or both).

Lemma 2. *If $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ has a fooling set of size t then $\chi(f) \geq t$ and $D(f) \geq \log_2 t$.*

Proof. Let F be a fooling set of size t for the function f. Consider for some $(x,y) \neq (x',y') \in F$, by definition of fooling set, we have

1. $f(x,y) = f(x',y')$ and

2. $f(x,y') \neq f(x,y)$ or $f(x',y) \neq f(x,y)$

Now consider a combinatorial rectangle that contains both $f(x,y)$ and $f(x',y')$, by definition of a combinatorial rectangle, both $f(x',y)$ and $f(x,y')$ must be in the same rectangle. However, at least one of $f(x',y)$ and $f(x,y')$ must have a different value from $f(x,y)$ and $f(x',y')$. Thus, for any pair of $(x,y) \neq (x',y') \in F$, a rectangle containing them both cannot be f-monochromatic.

Since we have a fooling set of size t, and every element inside the fooling set cannot share f-monochromatic rectangles with each other. Thus, we need at least t f-monochromatic rectangles to partition $\mathcal{X} \times \mathcal{Y}$, in other words, $\chi(f) \geq t$.

Furthermore, by **Lemma 1** from **Chapter 1**, $D(f) \geq \log_2 \chi(f)$. And since $\chi(f) \geq t$, we have $D(f) \geq \log_2 t$. □

______________________________________ 0

Exercise 6. $\chi(f) \geq t + 1$ if f is not constant.

2.2 Equality II

Using the fooling set bound, we can get an optimal lower bound on the communication complexity of the equality function.

Theorem 5. $D(\textsc{Eq}) \geq n$.

Proof. Consider the rectangle for $\textsc{Eq} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$.
Let $F = \{(x,x) : x \in \{0,1\}^n\}$.

Now, for any pair $x \neq y \in \{0,1\}^n$, consider (x,x) and $(y,y) \in F$, we have

1. $\textsc{Eq}(x, x) = \textsc{Eq}(y, y) = 1$
2. $\textsc{Eq}(x, y) = \textsc{Eq}(y, x) = 0 \neq 1$

Thus, F is a fooling set for $\textsc{Eq}$and we have $|F| = |\{0, 1\}^n| = 2^n$.

By **Lemma 1**, we have $D(\textsc{Eq}) \geq \log_2 |F| = \log_2 2^n = n$ □

Exercise 7. Prove that in fact $D(\textsc{Eq}) = n + 1$.

2.3 Set disjointness

The *set disjointness* function $\textsc{Disj} : 2^{[n]} \times 2^{[n]}$ is defined by

$$\textsc{Disj}(S, T) = \begin{cases} 1 & \text{if } S \cap T = \emptyset \\ 0 & \text{if } S \cap T \neq \emptyset. \end{cases}$$

Use the fooling set method to obtain optimal bounds on the communication complexity of the set disjointness function.

Theorem 6. *$D(\textsc{Disj}) = \Theta(n)$.*

Proof. For upper bound, by **Theorem 2** from **Chapter 1**, we know $D(\textsc{Disj})) \leq \log_2 |2^{[n]}| = n$. So $D(\textsc{Disj}) = O(n)$.

For lower bound, we will use the fooling set method.

Consider the rectangle for $\textsc{Disj} : 2^{[n]} \times 2^{[n]} \to \{0, 1\}$.
Let $F = \{(x, \overline{x}) : x \in 2^{[n]}\}$, and $\overline{x}$ is the complement of x.

Now, for any pair $x \neq y \in 2^{[n]}$, consider $(x, \overline{x})$ and $(y, \overline{y}) \in F$, we have

1. $\textsc{Disj}(x, \overline{x}) = \textsc{Disj}(y, \overline{y}) = 1$
   proof: A set is always disjoint with its complement
2. Either $\textsc{Disj}(x, \overline{y}) = 0$ or $\textsc{Disj}(\overline{x}, y) = 0$
   proof: At least one of x and y must contain an element that the other doesn't since $x \neq y$. Without loss of generality, assume x has an element a which y doesn't. Then $\overline{y}$ must contain a and so does x, thus $\textsc{Disj}(x, \overline{y}) = 0$.

By above claims, we conclude that F is a fooling set for $\textsc{Disj}$, and $|F| = |2^{[n]}| = 2^n$.
And by **Lemma 1**, we have $D(\textsc{Disj}) \geq \log_2 |F| = \log_2 2^n = n$, in other words, $D(\textsc{Disj}) = \Omega(n)$
Since $D(\textsc{Disj}) = O(n)$ and $D(\textsc{Disj}) = \Omega(n)$, we have $D(\textsc{Disj}) = \theta(n)$ □

2.4 Inner product

The fooling set bound is not always tight. One particularly noteworthy example where this method fails to give a good lower bound is the inner product function $\mathrm{IP} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ defined by

$$\mathrm{IP}(x,y) = \sum_{i=1}^{n} x_i y_i \pmod 2.$$

As we will see later, $D(\mathrm{IP}) = \Theta(n)$, but the fooling set bound can only give the much weaker bound of $D(\mathrm{IP}) = \Omega(\log n)$.

Theorem 7. *Every fooling set for the* IP *function has size at most* n^2*.*

Proof. Enter your proof here, if you complete it. □

2.5 Special case of the rectangle size bound

The partition bound says that the communication complexity of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is large if the minimum number of f-chromatic rectangles required to partition $\mathcal{X} \times \mathcal{Y}$ is large. And the number of f-chromatic rectangles required to partition $\mathcal{X} \times \mathcal{Y}$ must be large whenever the only f-chromatic rectangles are small. This observation is the core of the *rectangle size bound.*

Definition 9 ($m(f)$)**.** For a given function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, define the *maximum rectangle size* of f to be

$$m(f) = \max\{|R| : R \subseteq \mathcal{X} \times \mathcal{Y} \text{ is an } f\text{-monochromatic rectangle}\}.$$

Lemma 3. *For every function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$*,* $\chi(f) \geq \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)}$ *and therefore*

$$D(f) \geq \log_2 \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)}.$$

Proof. Since $m(f)$ is the size of largest f-monochromatic rectangle, all other possible f-monochromatic rectangles must have size smaller than or equal to $m(f)$. Thus, for any partition of $\mathcal{X} \times \mathcal{Y}$, the number of f-monochromatic rectangles must be at least $\frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)}$. In other words, $\chi(f) \geq \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)}$.

By **Lemma 1** from **Chapter 1**, $D(f) \geq \log_2 \chi(f)$, hence $D(f) \geq \log_2 \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)}$. □

2.6 Inner product II

Use the rectangle size bound to prove an optimal lower bound on the inner product function.

Theorem 8. $m(IP) \leq 2^n$ *and so* $D(IP) = \Theta(n)$*.*

Proof. Let R be the 0-rectangle of maximal size, and say $R = A \times B \in \mathcal{X} \times \mathcal{Y}$.

Now let $R' = \{A \cup \{0\}\} \times \{B \cup \{0\}\}$

Now if $IP(x,y) = 0$ and $IP(x',y) = 0$, then $IP(x + x', y) = 0$.

Similarly if $IP(x,y)=0$ and $IP(x,y')=0$, then $IP(x,y+y')=0$, and thus A, B are orthogonal subspace of $\mathbb{Z}_2^n$.

Hence $dim(A)+dim(B)\leq n$,
$2^{dim(A)+dim(B)}=|A||B|=m_0(\text{IP})\leq 2^n$.

□

2.7 Rectangle size bound

The (general) rectangle size bound is obtained by generalizing the observation from the last section: instead of just counting the number of elements in a rectangle, we can consider the measure of rectangles under *any* probability distribution on $\mathcal{X}\times\mathcal{Y}$.

Definition 10 ($m_\mu(f)$)**.** For a given function $f:\mathcal{X}\times\mathcal{Y}\to\{0,1\}$ and a given distribution μ on $\mathcal{X}\times\mathcal{Y}$, define the *maximum rectangle size of f with respect to μ* to be

$$m_\mu(f)=\max\{\mu(R):R\subseteq\mathcal{X}\times\mathcal{Y}\text{ is an }f\text{-monochromatic rectangle}\}.$$

Lemma 4. *For every function $f:\mathcal{X}\times\mathcal{Y}\to\{0,1\}$ and every distribution μ on $\mathcal{X}\times\mathcal{Y}$,*

$$\chi(f)\geq\frac{1}{m_\mu(f)}$$

and therefore

$$D(f)\geq\log_2\frac{1}{m_\mu(f)}.$$

Proof. By definition of $m_\mu(f)$, for any f-monochromatic rectangle R, $\mu(R)\leq m_\mu(f)$. So for any partition of $\mathcal{X}\times\mathcal{Y}$, the number of f-monochromatic rectangles is at least $\frac{1}{m_\mu(f)}$ since for any partition $\sum_R\mu(R)=1$ and any $\mu(R)\leq m_\mu(f)$. In other words, $\chi(f)\geq\frac{1}{m_\mu(f)}$.

Furthermore, By **Lemma 1** from **Chapter 1**, $D(f)\geq\log_2\chi(f)\geq\log_2\frac{1}{m_\mu(f)}$. □

2.8 Fooling sets and rectangle size bound

The fooling set bound is a special case of the rectangle size bound, as the following theorem shows.

Theorem 9. *If $f:\mathcal{X}\times\mathcal{Y}\to\{0,1\}$ has a fooling set $S\subseteq\mathcal{X}\times\mathcal{Y}$ of size $|S|=t$, then there is a distribution μ on $\mathcal{X}\times\mathcal{Y}$ for which $m_\mu(f)\leq 1/t$.*

Proof. Considering any fooling set S with size t and the following distribution μ:

$\mu:\mathcal{X}\times\mathcal{Y}\to\mathbb{R}_+$ is described as:
$\mu(s)=\frac{1}{t}$ for $s\in S$
$\mu(s)=0$ otherwise

Since by previous results we have at least t other f-monochromatic rectangles, and for each of these rectangles, say R':

-if it contains 0s, then $\mu(R') = 0$
-if it contains 1s, then $\mu(R') = \frac{1}{t}$ because no two elements in S can be in the same f-monochromatic rectangle.

Thus $m_\mu(f) \leq 1/t$. □

2.9 Log rank bound

Another convenient measure that lower bounds the partition number of a function is the log of the rank of the corresponding matrix.

Definition 11 (Matrix of a function)**.** The *matrix* M_f corresponding to the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is the $|\mathcal{X}| \times |\mathcal{Y}|$-dimensional $\{0,1\}$-valued matrix with rows indexed by $\mathcal{X}$ and columns indexed by $\mathcal{Y}$ defined by

$$(M_f)_{x,y} = f(x,y)$$

for every $x \in \mathcal{X}$ and $y \in \mathcal{Y}$.

Definition 12 (Rank)**.** The *rank* of the function f, denoted

$$\text{rank}(f),$$

is the linear rank of the matrix M_f over $\mathbb{R}$.

The logarithm of the rank of a function gives a lower bound on the communication complexity of the function.

Lemma 5. *For every* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$,

$$D(f) \geq \log_2 \text{rank}(f).$$

Proof. (Algebra)Consider $(M_f)_{x,y}$ which represents the rectangle $\mathcal{X} \times \mathcal{Y}$. In this matrix, we have $rank(f)$ rows that are linearly independent to each other which means for all $rank(f)$ rows, there exists some elements in it that cannot be represented by any linear combination of other rows. If such elements exists, then we need an extra rectangle for these elements in the process of partitioning. And since there are $rank(f)$ suck rows, we need at least $rank(f)$ f-monochromatic rectangles to partition $\mathcal{X} \times \mathcal{Y}$. In other words, $\chi(f) \geq rank(f)$.

By **Lemma 1** from **Chapter 1**, $D(f) \geq \log_2 \chi(f) \geq \log_2 rank(f)$.

(Combintorics) Let $R = \{R_i\}$ be a partition of M.
Then $\sum rank(R_i) \geq rank(R)$.

Let R be partition of f-monochromatic rectangles, then we have

$rank(R) \leq \sum_{R_i \in R} rank(R_i) \leq |R| = \chi(f)$.

□

Exercise 8. Give an alternative proof that $D(\textsc{Eq}) \geq n$ using the log rank bound.

Exercise 9. Give an alternative proof that $D(\text{IP}) \geq n$ using the log rank bound.

2.10 Greater than

The greater-than function $\text{GT} : [2^n] \times [2^n] \to \{0, 1\}$ is defined by

$$\text{GT}(x, y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{otherwise.} \end{cases}$$

Use the log rank bound to give an optimal lower bound on the greater-than function.

Theorem 10. *$D(\text{GT}) = \Theta(n)$.*

Proof. Consider Matrix of $\text{GT}(M_{\text{GT}})_{x,y}$. The matrix is filled with 1 for any index under the diagonal and 0 for any index above or on the diagonal, assuming x and y are sorted. It is obvious that the rank of $(M_{\text{GT}})_{x,y}$ is $2^n - 1$, and then by **Lemma 4**, we have $D(\text{GT}) \geq \log_2(2^n - 1)$, and $D(\text{GT}) = \Omega(n)$.

For the upper bound, by **Theorem 2** from **Chapter 1**, we know $D(\text{GT})) \leq \log_2 |[2^n]| = n$. So $D(\text{GT}) = O(n)$.

Thus, $D(\text{GT}) = \theta(n)$. □

2.11 Tightness of the log rank bound

Prove that the rank of a function can also be used to obtain an upper bound on the communication complexity of the function.

Theorem 11. *For every $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$, $D(f) \leq \text{rank}(f) + 1$.*

Proof. As we discussed in the proof of **Theorem 6**, if we have $rank(f)$ f-monochromatic rectangles, then we are able to represent all elements on the $rank(f)$ linearly independent rows.

For the rows other than these $rank(f)$, all of them can be written as a linear combination of some rows of the selected $rank(f)$ rows. This means the elements on these rows can be included in the previous f-monochromatic rectangles without introducing new ones. Thus, with these $rank(f)$ rectangles we will be able to contain all 1's in $\mathcal{X} \times \mathcal{Y}$, and an extra rectangle that represents 0s. Thus we need at most $rank(f) + 1$ f-monochromatic rectangles.

For the protocol tree, it has at most $rank(f) + 1$ leaves, and for the worst case, $height(T) = rank(f) + 1$. Thus $D(f) \leq rank(f) + 1$. □

Exercise 10. Show that there exists a function f for which $D(f) = \omega(\log_2 \operatorname{rank}(f))$.

Open Problem 2 (Log rank conjecture)**.** Prove that there exists a constant $c > 0$ such that every function f satisfies

$$D(f) = O(\log^c \operatorname{rank}(f)).$$

Chapter 3

Distributional communication complexity

Functions can become much easier to compute in the communication complexity setting when we allow the protocols to make some errors. We can define this formally using the notion of *distributional communication complexity.* We first explore this notion in its most natural setting, which corresponds to the uniform distribution.

Definition 13 (Protocol error)**.** Fix any $\epsilon \geq 0$. A protocol π *computes f with error at most ϵ under the uniform distribution* if it correctly outputs the value $f(x,y)$ for at least an $1-\epsilon$ fraction of all inputs, i.e., if

$$\Pr_{(x,y)}[\pi(x,y) \neq f(x,y)] \leq \epsilon$$

when (x,y) is drawn uniformly at random from $\mathcal{X} \times \mathcal{Y}$.

Definition 14 (Uniform distributional complexity)**.** For any $\epsilon \geq 0$, the *ϵ-error distributional communication complexity* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with respect to the uniform distribution,

$$D_\epsilon^{\text{unif}}(f),$$

is the minimum communication cost of a protocol that computes f with error at most ϵ under the uniform distribution.

Exercise 11. Show that $D_0^{\text{unif}}(f) = D(f)$ and that for every $\epsilon > 0$, $D_\epsilon^{\text{unif}}(f) \leq D(f)$.

Exercise 12. Show that every $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ satisfies $D_{1/2}^{\text{unif}}(f) = 0$.

3.1 Equality III

The ϵ-error distributional complexity of functions can sometimes be dramatically smaller than their deterministic complexity.

Theorem 12. *For any $\epsilon \geq \frac{1}{2^n}$, $D_\epsilon^{\text{unif}}(\text{EQUALITY}) = 0$.*

Proof. Consider a protocol p where Alice and Bob simply agrees on $\text{EQ}(Alice, Bob) = 0$ without sending any bits.

Now note that under uniform distribution, $Pr[x = y] = \frac{2^n}{2^n * 2^n} = \frac{1}{2^n}$. Thus $\epsilon = \frac{1}{2^n}$, and since we sent 0 bits, $D_\epsilon^{\text{unif}}(\text{EQUALITY}) = 0$. □

3.2 Uniform discrepancy

The *uniform discrepancy* of a function is a different way to measure how large "nearly f-monochromatic" rectangles can be. For this measure, it is convenient to represent the function with a ±1-valued (instead of $\{0,1\}$-valued) matrix.

Definition 15 (±1-Matrix of a function)**.** The *±1-matrix* $M_f^{\pm}$ corresponding to the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is the $|\mathcal{X}| \times |\mathcal{Y}|$-dimensional $\{-1,1\}$-valued matrix with rows indexed by $\mathcal{X}$ and columns indexed by $\mathcal{Y}$ defined by

$$\left(M_f^{\pm}\right)_{x,y} = (-1)^{f(x,y)}$$

for every $x \in \mathcal{X}$ and $y \in \mathcal{Y}$.

Definition 16 (Uniform discrepancy)**.** The *uniform discrepancy* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is

$$\mathrm{disc_u}(f) = \max_{A \subseteq \mathcal{X}, B \subseteq \mathcal{Y}} \frac{1}{|\mathcal{X}|\,|\mathcal{Y}|} \left| \sum_{x \in A, y \in B} \left(M_f^{\pm}\right)_{x,y} \right| .$$

We can use the uniform discrepancy to bound deterministic communication complexity.

Theorem 13. *For every function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$*,*

$$\chi(f) \geq \mathrm{disc_u}(f)^{-1}$$

and so $D(f) \geq \log_2 \frac{1}{\mathrm{disc_u}(f)}$*.*

Proof. Now think about the maximum f-monochromatic rectangle $R = a \times b$ in $\mathcal{X} \times \mathcal{Y}$.

$\left| \sum_{x,y \in R} \left(M_f^{\pm}\right)_{x,y} \right| = |R| = m(f)$ since every entry in R has the same value(-1 or 1).

Since that $\mathrm{disc_u}(f) \geq \frac{\left| \sum_{x,y \in R} \left(M_f^{\pm}\right)_{x,y} \right|}{|\mathcal{X}||\mathcal{Y}|} = \frac{m(f)}{|\mathcal{X}||\mathcal{Y}|}$, thus $\frac{1}{\mathrm{disc_u}(f)} \leq \frac{|\mathcal{X}||\mathcal{Y}|}{m(f)}$
and by **Lemma 2** from **Chapter 2**, we have

$\chi(f) \geq \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)} \geq \frac{1}{\mathrm{disc_u}(f)}$ and

$D(f) \geq \log_2 \frac{|\mathcal{X}|\,|\mathcal{Y}|}{m(f)} \geq \log_2 \frac{1}{\mathrm{disc_u}(f)}$,

□

3.3 Uniform discrepancy bound

We can also use uniform discrepancy to bound the uniform distributional communication complexity of functions.

Lemma 6. *For every function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ *and every* $0 \leq \epsilon < \frac{1}{2}$*,*

$$D_{\epsilon}^{\mathrm{unif}}(f) \geq \log_2 \left(\frac{1 - 2\epsilon}{\mathrm{disc_u}(f)} \right).$$

Proof. Since the protocol error is ϵ, we have that if we draw x and y uniformly at random:
$\Pr[\pi(x,y) \neq f(x,y)] \leq \epsilon$ and
$\Pr[\pi(x,y) = f(x,y)] \geq 1 - \epsilon$.
Thus $\Pr[\pi(x,y) = f(x,y)] - \Pr[\pi(x,y) \neq f(x,y)] \geq 1 - 2\epsilon$.

Now if we divide this value by each rectangle R in the protocol, we will have that
$\sum_R \Pr[\textit{select } x,y] * (\Pr_{(x,y \in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y \in R)}[\pi(x,y) \neq f(x,y)]) \geq 1 - 2\epsilon$
$\sum_R \Pr[\textit{select } x,y] * |\Pr_{(x,y \in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y \in R)}[\pi(x,y) \neq f(x,y)]| \geq 1 - 2\epsilon$

Note that the term $\Pr[\textit{select } x,y] * |\Pr_{(x,y \in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y \in R)}[\pi(x,y) \neq f(x,y)]|$ equals to $\frac{\left| \sum_{x,y \in R} \left(M_f^{\pm}\right)_{x,y} \right|}{|\mathcal{X}||\mathcal{Y}|}$ since we can use one of $(-1,1)$ to represent the correct value and the other for incorrect ones, and the probability of selecting each one is $\frac{1}{|\mathcal{X}||\mathcal{Y}|}$. Results will be the same since we are using absolute value.

Thus, we have that

$\sum_R \frac{\left|\sum_{x,y\in R}\left(M_f^{\pm}\right)_{x,y}\right|}{|\mathcal{X}||\mathcal{Y}|} \geq 1-2\epsilon$

$\sum_R \mathrm{disc_u}(f) \geq 1-2\epsilon$

Say there are α rectangles in the protocol, then $\alpha * \mathrm{disc_u}(f) \geq 1-2\epsilon$

$\alpha \geq \frac{1-2\epsilon}{\mathrm{disc_u}(f)}$

Then
$D_\epsilon^{\mathrm{unif}}(f) \geq \log_2\left(\frac{1-2\epsilon}{\mathrm{disc_u}(f)}\right)$, since there are at least $\frac{1-2\epsilon}{\mathrm{disc_u}(f)}$ rectangles in any protocol. □

3.4 Inner product III

The discrepancy method can be used to give optimal bounds on the distributional communication complexity of the inner product function over the uniform distribution.

Theorem 14. $\mathrm{disc_u}(\mathrm{IP}) \leq 2^{-n/2}$ *and, as a result,* $D_\epsilon^{\mathrm{unif}}(\mathrm{IP}) \geq \Omega(n)$ *for every* $\epsilon < \frac{1}{2}$.

Proof. Consider the ±1-Matrix of Inner Product function M_{ip}. Now calculate the value $M_{ip}M_{ip}^t$, we can see that the result is a matrix full of 1 since the inner product function is commutative which means $M_{ip(x,y)} = M_{ip(y,x)}$, thus every entry of M_{ip} is either $1*1$ or $-1*-1$. Since this is equivalent to the identity matrix, we conclude that M_{ip} is an orthogonal matrix.

Now consider any rectangle in $\mathcal{X} \times \mathcal{Y}$, say $R = a \times b$. We calculate the discrepancy of R, say this value is d.

$d = \frac{1}{|\mathcal{X}||\mathcal{Y}|}|\sum_{x\in a,y\in b}\left(M_{ip}\right)_{x,y}|$

$= \frac{1}{2^n * 2^n} 1_a^t M_{ip} 1_b$

$\leq \frac{1}{2^{2n}} \|1_a\| \, \|M_{ip}1_b\|$

$= \frac{1}{2^{2n}}\sqrt{1_a 1_a}\sqrt{M_{ip}1_b M_{ip}1_b}$

$\leq \frac{1}{2^{2n}}\sqrt{2^n}\sqrt{M_{ip}1_b M_{ip}1_b}$

$= \frac{1}{2^{2n}}\sqrt{2^n}\sqrt{1_b^t M_{ip}^t M_{ip} 1_b}$

$\leq \frac{1}{2^{2n}}\sqrt{2^n}\sqrt{1_b^t 2^n I 1_b}$

$\leq \frac{1}{2^{2n}}\sqrt{2^n}\sqrt{2^n * 2^n}$

$= \frac{1}{2^{n/2}} = 2^{-n/2}$

Since discrenpancy of any rectangle $\leq 2^{-n/2}$, we conclude that $\text{disc}_{\text{u}}(IP) \leq 2^{-n/2}$, and **Lemma 1**, $D_\epsilon^{\text{unif}}(\text{IP}) \geq \log_2(\frac{1-2\epsilon}{\text{disc}_{\text{u}}(IP)}) \geq \log_2(\frac{1-2\epsilon}{2^{-n/2}}) \geq \Omega(n)$ □

3.5 Distributional complexity

The definitions we have seen so far in this chapter all generalize to arbitrary distributions over the domain of the function.

Definition 17 (Protocol error)**.** Fix any $\epsilon \geq 0$. A protocol *computes* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ *with error at most* ϵ *under the distribution* μ *on* $\mathcal{X} \times \mathcal{Y}$ if it correctly outputs the value $f(x,y)$ for at least a $1-\epsilon$ measure of all inputs in μ, i.e., if

$$\Pr_{(x,y)\sim\mu}[\pi(x,y) \neq f(x,y)] \leq \epsilon.$$

Definition 18 (Distributional complexity)**.** For any $\epsilon \geq 0$, the ϵ*-error distributional communication complexity* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with respect to the distribution μ on $\mathcal{X} \times \mathcal{Y}$,

$$D_\epsilon^\mu(f),$$

is the minimum communication cost of a protocol that computes f with error at most ϵ under the distribution μ.

Definition 19 (μ-Discrepancy)**.** For any distribution μ on $\mathcal{X} \times \mathcal{Y}$, the μ*-discrepancy* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ is

$$\text{disc}_\mu(f) = \max_{A \subseteq \mathcal{X}, B \subseteq \mathcal{Y}} \left| \mu\big((A \times B) \cap f^{-1}(0)\big) - \mu\big((A \times B) \cap f^{-1}(1)\big) \right|.$$

Theorem 15. *For every function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ *and every distribution* μ *over* $\mathcal{X} \times \mathcal{Y}$,

$$\chi(f) \geq \text{disc}_\mu(f)^{-1}$$

and so $D(f) \geq \log_2 \frac{1}{\text{disc}_\mu(f)}$.

Proof. Consider the f-monochromatic rectangle $R = a \times b$ in $\mathcal{X} \times \mathcal{Y}$ with maximum size under distribution μ, like **Definition 3** from **Chapter 2** defined, we say the size of R is $m_\mu(f)$.

Now consider the μ-discrepancy of R, since the values in R are of the same, μ-discrepancy of R should be equal to $m_\mu(f)$. And we have $\text{disc}_\mu(f) \geq \mu$-discrepancy of $R \geq m_\mu(f)$.

By **Lemma 3** from **Chapter 2**, we have $\chi(f) \geq \frac{1}{m_\mu(f)} \geq \text{disc}_\mu(f)^{-1}$,

And $D(f) \geq \log_2 \frac{1}{m_\mu(f)} \geq \log_2 \text{disc}_\mu(f)^{-1}$

□

Exercise 13. Show that when μ is the uniform distribution over $\mathcal{X} \times \mathcal{Y}$, the definitions of μ-discrepancy and uniform discrepancy are equivalent.

Exercise 14. Show that for every distribution μ, $D_\epsilon^\mu(f) \leq D(f)$.

3.6 Discrepancy bound

The μ-discrepancy of a function can be used to give lower bounds on its μ-distributional communication complexity.

Lemma 7. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, every distribution μ on $\mathcal{X} \times \mathcal{Y}$, and every $0 \leq \epsilon < \frac{1}{2}$,*

$$D^{\mu}_{\epsilon}(f) \geq \log_2 \left(\frac{1-2\epsilon}{\mathrm{disc}_{\mu}(f)} \right).$$

Proof. Since the protocol error is ϵ, we have that if we draw x and y by a distribution μ:
$\Pr[\pi(x,y) \neq f(x,y)] \leq \epsilon$ and
$\Pr[\pi(x,y) = f(x,y)] \geq 1-\epsilon$.
Thus $\Pr[\pi(x,y) = f(x,y)] - \Pr[\pi(x,y) \neq f(x,y)] \geq 1-2\epsilon$.

Now if we divide this value by each rectangle R in the protocol, we will have that
$\sum_R \Pr[\mathit{select}\ x,y] * (\Pr_{(x,y\in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y\in R)}[\pi(x,y) \neq f(x,y)]) \geq 1-2\epsilon$
$\sum_R \Pr[\mathit{select}\ x,y] * |\Pr_{(x,y\in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y\in R)}[\pi(x,y) \neq f(x,y)]| \geq 1-2\epsilon$

Note that the term $\Pr[\mathit{select}\ x,y] * |\Pr_{(x,y\in R)}[\pi(x,y) = f(x,y)] - \Pr_{(x,y\in R)}[\pi(x,y) \neq f(x,y)]|$ equals to $\left|\sum_{x,y\in R} \mu(x,y) \big(M^{\pm}_f\big)_{x,y}\right|$ since we can use one of $(-1,1)$ to represent the correct value and the other for incorrect ones, and the probability of selecting each one is $\mu(x,y)$. Results will be the same since we are using absolute value.

Thus, we have that

$\sum_R \left|\sum_{x,y\in R} \mu(x,y) \big(M^{\pm}_f\big)_{x,y}\right| \geq 1-2\epsilon$

$\sum_R \mathrm{disc}_{\mu}(f) \geq 1-2\epsilon$

Say there are α rectangles in the protocol, then $\alpha * \mathrm{disc}_{\mu}(f) \geq 1-2\epsilon$

$\alpha \geq \frac{1-2\epsilon}{\mathrm{disc}_{\mu}(f)}$

Then
$D^{\mathrm{unif}}_{\epsilon}(f) \geq \log_2 \left(\frac{1-2\epsilon}{\mathrm{disc}_{\mu}(f)}\right)$, since there are at least $\frac{1-2\epsilon}{\mathrm{disc}_{\mu}(f)}$ rectangles in any protocol. □

3.7 Equality IV

Using the discrepancy bound, we can show that there are distributions for which the equality function requires Alice and Bob to exchange *some* bits of communication to obtain a non-trivial error.

Theorem 16. *There exists a distribution μ over $\{0,1\}^n \times \{0,1\}^n$ for which* $\mathrm{disc}_{\mu}(\textsc{Eq}) \leq \frac{1}{8} * \frac{2^n}{2^n-1}$ *and so*

$$D^{\mu}_{\epsilon}(\textsc{Eq}) \geq 1$$

for every $\epsilon \leq \frac{3}{8}$.

Proof. Consider the following distribution μ:

$\mu(x,y) = \frac{1/2}{2^n} = \frac{1}{2^{n+1}}$ if $EQ(x,y) = 1$,
$\mu(x,y) = \frac{1/2}{2^{2n}-2^n} = \frac{1}{2^{2n+1}-2^{n+1}}$ otherwise.

Now we calculate the μ-discrepancy of EQ.

First think of a rectangle containing total of i 1's, then by the structure of rectangles from $EQ(x,y)$, we know that it has at least $(i-1)i$ 0's. This gives the discrepancy of

$\frac{(i-1)i}{2^{2n+1}-2^{n+1}} - \frac{i}{2^{n+1}} = \frac{i(i-2^n)}{2^{n+1}(2^n-1)}$ Note the value is negative.

Since choosing $i = 2^{n-1}$ will make this term has the maximum absolute value, we have discrepancy

$\leq |\frac{2^{n-1}(2^{n-1}-2^n)}{2^{n+1}(2^n-1)}| = \frac{1}{4}|\frac{2^{n-1}-2^n}{2^n-1}| \leq \frac{1}{8}|\frac{2^n-2^{n+1}}{2^n-1}| \leq \frac{1}{8} * \frac{2^n}{2^n-1}$

Now think of a rectangle containing only 0's, the largest possible one should be of size $2^{n-1} * 2^{n-1}$ since the column and rows cannot share any index, and to maximize such a rectangle is to make it a square.

For this rectangle, the μ-discrepancy is easy to calculate:

$2^{n-1} * 2^{n-1} * \frac{1}{2^{2n+1}-2^{n+1}} = \frac{2^{2n-2}}{2^{2n+1}-2^{n+1}} = \frac{1}{8} * \frac{2^n}{2^n-1}$

Now consider any other rectangle, they must be a combination of the above two cases:

For the rows with 1, they form a square with potentially a tailing rectangle full of 0's. The rest of the rows forms a rectangle filled with 0's. Note that adding these discrepancy yield a smaller value since for the above two cases we have opposite signs. Thus for any rectangle in $EQ(x,y)$, we have μ-discrepancy $\leq \frac{1}{8} * \frac{2^n}{2^n-1}$. In other words, $\text{disc}_\mu(\text{EQ}) \leq \frac{1}{8} * \frac{2^n}{2^n-1}$. And plugging this result into **Lemma 2** yields $D^\mu_\epsilon(\text{EQ}) \geq 1$. □

3.8 Equality V

Show that the bound in the last section is essentially tight in that the distributional communication complexity of the equality function is constant when ϵ is a positive constant.

Theorem 17. *For every distribution* μ *over* $\{0,1\}^n \times \{0,1\}^n$,

$$D^\mu_{1/4}(\text{EQ}) = O(1).$$

Proof. If Alice and Bob shares randomness, then we can use the following protocol:

Alice and Bob agree on two randomly selected subsets of n-bit binary string, then Alice XORs bits from each subset together. Then send the two results to Bob. Bob calculate the sum of same subsets and compare them to the results Alice sent, then:

If both bits equal to Alice's, return 1 to Alice

Otherwise, return 0 to Alice.

Note this protocol sends only 3 bits, so the distribution communication complexity is $O(1)$. Now we calculate the error probability:

Let $h_s(x)$ be one of the result from Alice using subset$s$, $h_s(y)$ be the result from same subset from Bob. Assume $x \neq y$.
Let E be the event where $h_s(x) = h_s(y)$, let F be the event where $h_{s\ without\{i\}}(x) = h_{s\ without\{i\}}(y)$.

$Pr[E] = Pr[F] * Pr[E|F] + Pr[\overline{F}] * Pr[E|\overline{F}]$

$= Pr[F] * Pr[i \in s] + Pr[\overline{F}] * Pr[i \in s]$

$= Pr[F] * \frac{1}{2} + Pr[\overline{F}] * \frac{1}{2} = \frac{1}{2}$

And the probability of both time fail is $\frac{1}{4}$. Since we have correct result with probability 1 when $x = y$ and probability $\frac{3}{4}$ when $x \neq y$, we have $\epsilon = \frac{1}{4}$.

Thus $D^{\mu}_{1/4}(\text{EQ}) = O(1)$. □

3.9 Corruption bound

The *corruption bound* is another powerful technique for proving lower bounds in distributional communication complexity.

Lemma 8. *Fix a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ and a distribution μ on $\mathcal{X} \times \mathcal{Y}$. If there exist parameters $\alpha, \beta > 0$ for which every rectangle $R \subseteq \mathcal{X} \times \mathcal{Y}$ satisfies*

$$\mu(R \cap f^{-1}(0)) \geq \alpha \cdot \mu(R) - \beta$$

then

$$D^{\mu}_{\epsilon}(f) \geq \log\left(\frac{\alpha \cdot \mu(f^{-1}(1)) - \epsilon}{\beta}\right).$$

Proof. First we sum up $\mu(R \cap f^{-1}(0))$ for all rectangles R in a protocol:

$\sum_R \mu(R \cap f^{-1}(0))$

$= \mu(f^{-1}(0))$

$\geq \sum_R \alpha \cdot \mu(R) - \beta$

$= \sum_R \alpha \cdot (\mu(R \cap f^{-1}(0)) + \mu(R \cap f^{-1}(1))) - \beta$ if there are d rectangles

$= \alpha\mu(f^{-1}(0)) + \alpha\mu(f^{-1}(1)) - d * \beta$

Now since we have $\mu(f^{-1}(0)) \geq \alpha\mu(f^{-1}(0)) + \alpha\mu(f^{-1}(1)) - d * \beta$

$d \geq \frac{(\alpha-1)\mu(f^{-1}(0))+\alpha\mu(f^{-1}(1))}{\beta}$

$\geq \frac{\alpha\mu(f^{-1}(1))}{\beta}$ since $\alpha \leq 1$

$\geq \frac{\alpha\mu(f^{-1}(1))-\epsilon}{\beta}$

And since the number of rectangles is at least $\frac{\alpha\mu(f^{-1}(1))-\epsilon}{\beta}$, $D^{\mu}_{\epsilon}(f) \geq \log\left(\frac{\alpha\cdot\mu(f^{-1}(1))-\epsilon}{\beta}\right)$. □

3.10 Set Disjointness II

The corruption bound can be used to prove a strong lower bound on the distributional communication complexity of the set disjointness function. The key claim that is used to obtain this result is the following combinatorial statement.

Proposition 1. *For every $\alpha > 0$, there exists a $\delta > 0$ such that every rectangle $R = A \times B \subseteq 2^{[n]} \times 2^{[n]}$ that satisfies*

$$\Pr_{(S,T)\in R}[S \cap T = \emptyset] \geq 1 - \alpha$$

has size bounded by

$$|A| \leq 2^{-\delta\sqrt{n}}\binom{n}{\sqrt{n}} \qquad \textit{or} \qquad |B| \leq 2^{-\delta\sqrt{n}}\binom{n}{\sqrt{n}}.$$

Use the claim to complete the lower bound on the communication complexity of the set disjointness function.

Theorem 18. *Let μ be the uniform distribution on pairs $(S,T) \in 2^{[n]} \times 2^{[n]}$ that satisfy $|S| = |T| = \sqrt{n}$. Then*

$$D^{\mu}_{\epsilon}(\textsc{Disj}) \geq \Omega(\sqrt{n}).$$

Proof. First we consider the value of $\mu(R \cap f^{-1}(0))$:

$\mu(R \cap DISJ^{-1}(0)) = \mu(R) \cdot Pr[DISJ(x,y) = 0|x,y \in R] \geq \mu(R) \cdot \alpha \geq \alpha \cdot \mu(R) - \beta$ for any $\beta > 0$ if we use the same α for both equations.

Thus, by **Lemma 3**, we have that $D^{\mu}_{\epsilon}(DISJ) \geq \log\left(\frac{\alpha\cdot\mu(DISJ^{-1}(1))-\epsilon}{\beta}\right)$.

By **Proposition 1** we have that $|S| = |T| = \sqrt{n}$, so we can conclude that

$D^{\mu}_{\epsilon}(DISJ) \geq \log\frac{\alpha\cdot 2^{\sqrt{n}}-\epsilon}{\beta} = \Omega(\sqrt{n})$ □

Chapter 4

Randomized communication complexity

Until now, we have been focused on deterministic communication protocols. In this chapter, we study *randomized* communication protocols. We first focus on what is known as the *public-randomness model* of communication.

Definition 20 (Randomized protocol)**.** A *randomized communication protocol* Π is a distribution over deterministic communication protocols. The *(worst-case) cost* of Π is the maximum cost of any protocol in its support. The randomized protocol Π *computes* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with error at most ϵ if for every $(x,y) \in \mathcal{X} \times \mathcal{Y}$,

$$\Pr_{\pi \sim \Pi}[\pi(x,y) \neq f(x,y)] \leq \epsilon.$$

Definition 21 (Randomized communication complexity)**.** For $0 < \epsilon < \frac{1}{2}$, the *randomized communication complexity* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ at error ϵ is

$$R_\epsilon(f) := \min_{\Pi} \operatorname{cost}(\Pi)$$

where the minimum is taken over all randomized communication protocols that compute f with error at most ϵ.

Remark 1. The notation for randomized communication complexity can vary. In Kushilevitz and Nisan, for instance, the randomized communication complexity defined above is written as R_ϵ^{pub} and our symbol R_ϵ is reserved for another model of randomized communication (namely: the private randomness model we will see at the end of the chapter).

Exercise 15. Another way to define randomized communication protocols is to consider a model where Alice and Bob have access to their respective inputs *and* to a common sequence of coins that are flipped at random. Show that this model is equivalent to the one described above.

Exercise 16. Show that with the definition above, $R_0(f) = D(f)$ always holds. For this reason, $R_0(f)$ is usually reserved to the randomized model of communication where the cost of Π on input (x,y) is defined to be the *average* cost of the protocols in the support of Π.

4.1 Randomized and distributional complexity

Lower bounds in randomized communication complexity are often (usually?) obtained by establishing the lower bounds in the distributional communication complexity model instead. This approach is justified by the following relation, which is sometimes referred to as *(the easy direction of) Yao's minimax principle.*

Theorem 19. *For every $0 \leq \epsilon \leq \frac{1}{2}$, every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, and every distribution μ over $\mathcal{X} \times \mathcal{Y}$,*

$$R_\epsilon(f) \geq D_\epsilon^\mu(f).$$

Proof. Assume we have a protocol has distributional communication complexity of $D_\epsilon^\mu(f)$ for some distribution μ. This means that the protocol is able to compute f with error probability ϵ under distribution μ with run time $D_\epsilon^\mu(f)$.

Now by way of contradiction, assume $R_\epsilon(f) < D_\epsilon^\mu(f)$ for some f and μ. Then there exists an protocol inside the distribution that computes f under μ faster than $D_\epsilon^\mu(f)$. However, this means that this deterministic protocol is faster than the lower bound of all deterministic protocol that compute f. So by contradiction, $R_\epsilon(f) \geq D_\epsilon^\mu(f)$. □

4.2 Yao's minimax principle

Yao's minimax principle says something quite a bit stronger than the bound seen in the last chapter: it says that the randomized communication complexity of a function is *equal* to the maximum distributional complexity of the function over any distributions on its domain.

Theorem 20. *For every $0 \leq \epsilon \leq \frac{1}{2}$ and every function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$,*

$$R_\epsilon(f) = \max_\mu D^\mu_\epsilon(f)$$

where the maximum is taken over all distributions on $\mathcal{X} \times \mathcal{Y}$.

Proof. (Method1)First consider the protocol π that computes f with run time $t = \max_\mu D^\mu_\epsilon(f)$ under some distribution μ'.

Then since randomized communication protocol is a distribution over deterministic communications, consider the randomized communication protocol where the distribution is simply using π with probability 1. The run time will be less or equal to t under any distribution since t was taken maximum over all distributions.

Thus, since $R_\epsilon(f)$ was taken minimum among all distributions of deterministic protocols, $R_\epsilon(f) \leq \max_\mu D^\mu_\epsilon(f)$. Combining this result with **Theorem 1**, $R_\epsilon(f) = \max_\mu D^\mu_\epsilon(f)$.

(Method2)Let Π be a protocol computing f with $R_\epsilon(f)$, and let R be the public randomness used by Π, then for any input (x, y) we have that

$\mathbb{1}(\Pi_R(x, y) \neq f(x, y)) \leq \epsilon$
dist ζ.
$\mathbb{E}_{(x,y)\sim\zeta}\ \mathbb{E}_R\ \mathbb{1}(\Pi_R(x, y) \neq f(x, y)) \leq \epsilon$
$\mathbb{E}_R\ \mathbb{E}_{(x,y)\sim\zeta}\ \mathbb{1}(\Pi_R(x, y) \neq f(x, y)) \leq \epsilon$
$\mathbb{E}_{(x,y)\sim\zeta}\ \mathbb{1}(\Pi_R(x, y) \neq f(x, y)) \leq \epsilon$

□

4.3 Error reduction

The randomized communication complexity functions is usually studied in the setting where $\epsilon = \frac{1}{3}$. In fact, this is so common that shorthand notation is used for that case.

Definition 22. The *(two-sided error) randomized communication complexity* of $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ is

$$R(f) = R_{1/3}(f).$$

The reason most of the work focuses on this particular choice of ϵ is that a general transformation can be used to bound $R_\epsilon(f)$ for any $\epsilon > 0$ as a function of $R(f)$. This result is sometimes known as the *(majority) confidence amplification* trick.

Theorem 21. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ and every $0 < \epsilon < \frac{1}{2}$,*

$$R_\epsilon(f) = O\left(R(f) \cdot \log \tfrac{1}{\epsilon}\right).$$

Proof. Consider the following way to reduce error:

we run the protocol f for i times, then it returns 1 if more than half results are 1 and vice versa.

Now by Hoeffding's Inequality, $Pr[\frac{1}{m}\sum_{i=1}^{m} X_i \geq \frac{1}{2}] \leq e^{-\frac{m}{18}}$.

To bound this error probability below ϵ, we should have

$e^{-\frac{m}{18}} \leq \epsilon$
$-\frac{m}{18} \leq \ln \epsilon$
$m \geq -18 \ln \epsilon$
$m \geq 18 \ln \frac{1}{\epsilon} = O(\log \frac{1}{\epsilon})$

So if we repeat the protocol with error probability $\frac{1}{3}$ for $O(\log \frac{1}{\epsilon})$ times, we will have error probability of ϵ, which gives run time $O\left(R(f) \cdot \log \frac{1}{\epsilon}\right)$. □

Exercise 17. Extend the result in the theorem to show that for every $\delta > 0$, we also have

$$R_\epsilon(f) = O\left(R_{\frac{1}{2}-\delta}(f) \cdot \phi(\epsilon, \delta)\right)$$

for some appropriate function ϕ on ϵ and δ.

4.4 Equality VI

Theorem 22. *For every* $0 < \epsilon < \frac{1}{2}$,

$$R_\epsilon(\mathrm{EQ}) = O\left(\log \tfrac{1}{\epsilon}\right).$$

Proof. Using the same protocol and proof from Question **Equality V**, we have that $D^{\mu}_{1/4}(EQ) = O(1)$.

By Yao's Minimax Principle, we also have $R_{1/4}(EQ) = O(1)$, and since $1/4 \leq 1/3$, $R(f) = O(1)$

By **Theorem 3**, we have $R_\epsilon(EQ) = O(R(EQ) \cdot \log \frac{1}{\epsilon}) = O(\frac{1}{\epsilon})$. □

4.5 Greater than II

Recall that the function $\mathrm{GT} : [2^n] \times [2^n] \to \{0, 1\}$ is defined by

$$\mathrm{GT}(x, y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{otherwise.} \end{cases}$$

We saw in Chapter 2 that the deterministic communication complexity of this function is $\Theta(n)$. Its randomized communication complexity is much smaller.

Theorem 23. *The randomized communication complexity of* $\mathrm{GT} : [2^n] \times [2^n] \to \{0,1\}$ *is bounded above by*

$$R(\mathrm{GT}) = O\left(\log^2 n\right).$$

Proof. Note that the problem of GT is actually looking for the first bit where x and y differs. Now consider the following protocol:

```
If the length of current bit string is 1, compare the two bits and return the result
Run EQUALITY on the first half of $x$ and $y$
If they are equal, repeat the same protocol for the second half of $x$ and $y$
Otherwise, repeat the same protocol for the first half of $x$ and $y$
```

The algorithm is basically a binary search for the first bit where x and y differs using EQUALITY function and the protocol will return the correct value if all EQUALITY function returns correct value.

In other words, this happens when no EQUALITY function returns false value, say each with probability ϵ. By Union Bound, the probability will be less or equal to $\log n \cdot \epsilon \leq \frac{1}{3}$. If we want to achieve this inequality, we need to have $\epsilon\frac{1}{3\log n}$, and by **Theorem 3**, this require us to run EQUALITY for $\log 3 \log n$ times which gives run time $O(\log\log n)$ for each round. This gives us a run time of $O(\log n \log\log n)$ with less than or equal to $\frac{1}{3}$ error probability.

Thus, $R(\mathrm{GT}) = O\left(\log n \log\log n\right)$. □

Exercise 18. Improve the upper bound to show $R(\mathrm{GT}) = O\left(\log n \log\log n\right)$.

Exercise 19. Prove that $R(\mathrm{GT}) = \Theta\left(\log n\right)$.

4.6 Hamming distance

The *k-Hamming distance* function $\mathrm{HD}_k : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is defined by

$$\mathrm{HD}_k(x,y) = \begin{cases} 1 & \text{if } |\{i \in [n] : x_i \neq y_i\}| \leq k \\ 0 & \text{otherwise.} \end{cases}$$

Theorem 24. *For every* $1 \leq k \leq \frac{n}{2}$,

$$R(\mathrm{HD}_k) = O(k \log k).$$

Proof. Consider following protocol:

Make $8k^2$ groups, insert each index in $[n]$ into one of the $8k^2$ groups chosen uniformly at random. This result is shared by Alice and Bob.

For each group, Alice and Bob add all bits from the indices inside together.

For the result binary string, Alice and Bob use binary search and EQUALITY to find the groups that differ. Remove such a bit if it's found. Repeat this $k+1$ times or until $x = y$.

If the protocol didn't reach the $k+1$th step, return 1, otherwise returns 0.

The run time is clearly $O(klogk^2) = O(klogk)$.

Now we calculate the probability that two index are in the same group where bits from$x$ and y differ at both:

$Pr[Two\ indices\ are\ in\ the\ same\ group] = \frac{1}{8k^2} \cdot \frac{1}{8k^2} \cdot 8k^2 = \frac{1}{8k^2}$

And by union bound, this does not happen to any pair of indices is $\binom{n}{2}\frac{1}{8k^2} \leq \binom{n}{2}\frac{1}{2n^2} \leq \frac{1}{4}$, it gives that any pair of groups either differ by one or zero elements. Thus they must have different parity, and the total number of difference will be sum of these bits as a real number.

□

4.7 Private randomness

In the beginning of the chapter, we saw that the (public-coin) randomized communication model corresponds to the model where some coins are flipped and both Alice and Bob see the outcome of those coin flips. It is natural to study the alternative model where Alice and Bob both have access to coins they can flip (or, more abstractly, to some sources of randomness), but they each only see the outcome of their own coin flips. This model of communication corresponds to the *private-coin randomized communication* model.

Definition 23. A *private-coin randomized communication protocol* π is equivalent to a deterministic communication protocol with the two additions:

1. Each of Alice's internal node v is labelled with a function $h_v : \mathcal{X} \times R_A \to \{0,1\}$ and each of Bob's internal node w is labelled with a function $h_v : \mathcal{Y} \times R_B \to \{0,1\}$;
2. Alice has a distribution μ_A over R_A and Bob has a distribution μ_B over R_B.

The *(worst-case) cost* of Π is the maximum cost of any protocol in its support. The randomized protocol Π *computes* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with error at most ϵ if for every $(x,y) \in \mathcal{X} \times \mathcal{Y}$,

$$\Pr_{r_a \sim \mu_A, r_b \sim \mu_B}[\pi(x,y) \neq f(x,y)] \leq \epsilon.$$

Definition 24 (Private-coin randomized communication complexity)**.** For $0 < \epsilon < \frac{1}{2}$, the *private coin randomized communication complexity* of $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ at error ϵ is

$$R_\epsilon^{\text{priv}}(f) := \min_{\Pi} \text{cost}(\Pi)$$

where the minimum is taken over all private-coin randomized communication protocols that compute f with error at most ϵ.

Remark 2. As in the public-coin setting, we write $R^{\text{priv}}(f) := R^{\text{priv}}_{1/3}(f)$.

Exercise 20. Show that for every function f and every parameter $\epsilon > 0$, $R_\epsilon(f) \le R^{\text{priv}}_\epsilon(f)$.

Exercise 21. Show that error amplification also holds in the private coin model: for every $\epsilon > 0$, $R^{\text{priv}}_\epsilon(f) = O(R^{\text{priv}}(f) \log \frac{1}{\epsilon})$.

4.8 Newman's theorem

The gap between the public-coin and private-coin randomized communication complexities of a function cannot be arbitrarily large.

Theorem 25. *For every distribution μ over $\{0,1\}^n \times \{0,1\}^n$ and every constant $\epsilon > 0$,*

$$R^{\text{priv}}_{2\epsilon}(f) \le R_\epsilon(f) + O\left(\log \frac{n}{\epsilon^2}\right).$$

Proof. Consider we drawn $t = n/\epsilon^2$ random strings independently at random.

Now for the public coin protocol using these strings, the error probability is smaller than or equal to ϵ for every (x, y). We fix any pair of (x, y) consider the probability that 2ϵ of the t giving incorrect output for (x, y):

By Chernoff Bound and $\delta = \frac{\epsilon}{1-\epsilon} \le \epsilon$,

$Pr[X \le (1-\delta)\mu] \le e^{-\frac{\delta^2\mu}{2}} = e^{-\frac{\frac{\epsilon^2}{(1-\epsilon)^2}(1-\epsilon)t}{2}}$

Since we set $t = n/\epsilon^2$,

$Pr[X \le (1-\delta)\mu] \le e^{-\frac{\frac{\epsilon^2}{(1-\epsilon)^2}(1-\epsilon)n/\epsilon^2}{2}} = e^{O(n)}$

Now by Union bound,
$Pr[error\ probability\ >\ 2\epsilon] \le \binom{t}{2\epsilon t} e^{O(n)} \le \frac{1}{4}$ for some O(n)

Since in all the t select strings, at most 2ϵ gives incorrect results, Alice can just draw from these t strings uniformly at random and send it to Bob, which takes $O(\log \frac{n}{\epsilon^2})$ extra time. Then Alice and Bob have some shared randomness and thus be able to compute f with $R_\epsilon(f)$ run time. □

Remark 3. *Hint.* First consider what happens when you run a public-coin randomized protocol on $t = n/\epsilon^2$ random strings drawn independently at random. Then Chernoff bounds and the probabilistic method may be useful in completing the proof.

4.9 Private-coin randomness and determinism

Interestingly (and unlike in the public-coin model), the gap between the private-coin randomized and deterministic communication complexities of a function also cannot be arbitrarily large.

Theorem 26. *For every function* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$,

$$R^{\text{priv}}(f) = \Omega(\log D(f)).$$

Proof. Consider the communication setting where Alice and Bob can send real numbers with cost O(1):

There exists a private-coin randomized protocol with run time $t = R^{\text{priv}}(f)$ that computes f with error probability less than or equal to 1/3.

Since we are able to send real numbers in O(1) time, then we know the private-coin randomized protocol sent t real numbers in order to ensure an error probability of less than or equal to 1/3.

□

Remark 4. *Hint.* It might be easiest to first consider a communication setting where Alice and Bob can send real numbers with cost $O(1)$ and show that in this setting there is a deterministic protocol that computes f with cost $O(2^{R^{\text{priv}}(f)})$.

4.10 Equality VII

We can use the results obtained in the previous sections to obtain tight bounds on the private-coin randomized communication complexity of the equality function.

Theorem 27. *The private-coin randomized communication complexity of the* $\text{E}\textsc{q} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *function is*

$$R^{\text{priv}}(\text{E}\textsc{q}) = \Theta(\log n).$$

Proof. First by **Theorem 8**, $R^{\text{priv}}(\text{E}\textsc{q}) = \Omega(\log n)$.

Second by **Newman's Theorem**, $R^{\text{priv}}_{2/3}(f) \leq R(EQ) + O\left(\log \frac{n}{\epsilon^2}\right) = O(logn)$

And since error reduction works in private-coin randomized protocol as well, we can use constant factor of run time to reduce the error probability to 1/3.

Thus, $R^{\text{priv}}(\text{E}\textsc{q}) = \Theta(\log n)$. □

Exercise 22. Prove the upper bound $R^{\text{priv}}(\text{E}\textsc{q}) = O(\log n)$ directly without using Newman's theorem.

Chapter 5

Information complexity

Communication complexity is concerned mostly with the *minimum number of bits* that Alice and Bob need to exchange in order to compute some function on their joint inputs. Information complexity, on the other hand, is concerned with the *minimum amount of information* contained in the bits that Alice and Bob exchange to compute that function. Our first task is to formally define this notion of information. To do so, we will use standard definitions from information theory.

Definition 25 (Entropy)**.** The *entropy* of a random variable Z drawn from the probability distribution μ over a finite set $\mathcal{Z}$ is

$$H(Z) = -\sum_{z \in \mathcal{Z}} \mu(z) \log_2 \mu(z).$$

The entropy of a sequence of random variables $Z_1, Z_2, \ldots, Z_k$, denoted $H(Z_1\, Z_2 \cdots Z_k)$ is the entropy of the random variable $Z = (Z_1, Z_2, \ldots, Z_k)$.

Definition 26 (Conditional entropy)**.** The *conditional* entropy of Z given another random variable Z' is

$$H(Z \mid Z') = H(Z\, Z') - H(Z').$$

Definition 27 (Mutual information)**.** The *mutual information* of two random variables Z and Z' is

$$I(Z; Z') = H(Z) - H(Z \mid Z').$$

The *conditional mutual information* of Z and Z' given W is

$$I(Z; Z' \mid W) = H(Z \mid W) - H(Z \mid Z'\, W).$$

We use the following basic properties of entropy and mutual information.

Theorem 28. *Entropy satisfies the following properties:*

Boundedness *For every Z over a finite set $\mathcal{Z}$, $0 \le H(Z) \le \log_2 |\mathcal{Z}|$.*

Chain rule $H(Z_1\, Z_2 \cdots Z_k) = H(Z_1) + H(Z_2 \mid Z_1) + \cdots + H(Z_k \mid Z_1 \cdots Z_{k-1})$.

Subadditivity $H(Z \mid Z') \le H(Z)$ *and* $H(Z\, Z') \le H(Z) + H(Z')$.

Mutual information satisfies the following properties:

Boundedness $0 \le I(Z; Z') \le \min\{H(Z), H(Z')\}$.

Chain rule $I(Z_1\, Z_2; W) = I(Z_1; W) + H(Z_2; W \mid Z_1)$.

Data processing inequality *Whenever $Z' = f(Z)$ is determined by Z, $I(Z'; W) \le I(Z; W)$.*

5.1 External information complexity

Throughout this chapter, we will consider randomized protocols Π that have access to both public- and private-coin randomness.

Definition 28 (Transcript)**.** The *transcript* of a protocol Π on some input $(x, y) \in \mathcal{X} \times \mathcal{Y}$, which we will denote by

$$\Gamma^{\Pi}_{x,y}$$

is a bit string that includes (i) the public-coin random string R used by Alice and Bob and (ii) the sequence of bits that they exchange.

The first natural notion of information of a protocol that we will study is the amount of information that an external observer learns about Alice and Bob's input (x, y) by seeing the transcript $\Gamma^{\Pi}_{x,y}$ of their communication protocol.

Definition 29 (External information cost)**.** The *external information cost* of a randomized protocol Π over the distribution μ on $\mathcal{X} \times \mathcal{Y}$ is

$$\mathrm{icost}_{\mu}^{\mathrm{ext}}(\Pi) = I\big(X\,Y; \Gamma_{X,Y}^{\Pi}\big)$$

where $(X, Y) \sim \mu$.

Definition 30 (External information complexity)**.** The *ϵ-error external information complexity* of the function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ with respect to the distribution μ is

$$\mathrm{IC}_{\mu,\epsilon}^{\mathrm{ext}}(f) = \inf_{\pi} \mathrm{icost}_{\mu}^{\mathrm{ext}}(\pi)$$

with the infimum taken over all randomized protocols that compute f with error at most ϵ.[1]

The external information complexity of a function gives a lower bound on its randomized communication complexity.

Theorem 29. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ and every distribution μ on $\mathcal{X} \times \mathcal{Y}$,*

$$\mathrm{IC}_{\mu,\epsilon}^{\mathrm{ext}}(f) \leq R_{\epsilon}(f).$$

Proof. By boundedness of mutual information, $0 \leq I(Z; Z') \leq \min\{H(Z), H(Z')\}$, thus for any $\mathrm{icost}_{\mu}^{\mathrm{ext}}(\Pi) = I\big(X\,Y; \Gamma_{X,Y}^{\Pi}\big)$, we have that $\mathrm{icost}_{\mu}^{\mathrm{ext}}(\Pi) \leq \min\{H(X\,Y), H(\Gamma_{X,Y}^{\Pi})\}$.

Since $H(\Gamma_{X,Y}^{\Pi}) \leq |\Gamma_{X,Y}^{\Pi}|$ by boundedness of entropy, we have $\mathrm{icost}_{\mu}^{\mathrm{ext}}(\Pi) \leq |\Gamma_{X,Y}^{\Pi}|$.

And since the size of $|\Gamma_{X,Y}^{\Pi}|$ is the total bits exchanged by the protocol, we have that $\mathrm{icost}_{\mu}^{\mathrm{ext}}(\Pi) \leq R_{\epsilon}(f)$ for any protocol π. Thus $\mathrm{IC}_{\mu,\epsilon}^{\mathrm{ext}}(f) \leq R_{\epsilon}(f)$. □

5.2 Public randomness can be eliminated

When designing protocols to obtain upper bounds on the information complexity of functions, it is convenient to work in the framework where Alice and Bob can use both public- and private-coin randomness. For lower bounds, however, it is useful to note that without loss of generality we can consider protocols that use only private randomness.

Theorem 30. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$, every distribution μ over $\mathcal{X} \times \mathcal{Y}$, and every $0 \leq \epsilon \leq \frac{1}{2}$,*

$$\mathrm{IC}_{\mu,\epsilon}^{\mathrm{ext}}(f) = \inf_{\pi} \mathrm{icost}_{\mu}^{\mathrm{ext}}(\pi)$$

even when the infimum is taken only over private-coin randomized protocols that compute f with error at most ϵ.

Proof. Consider the case where only private randomness is allowed, then any private-coin protocol π can simulate public randomness by sending the private random bits over. In other words, all private random bits produced must be included inside the transcript comparing to public coin

[1] Note that the error of a randomized protocol is still defined to be its maximum error probability over *any* input in $\mathcal{X} \times \mathcal{Y}$; it is *not* the expected error over an input drawn from μ.

protocols.

However, since we assume the random bits are independent of the protocols inputs XY, sending the random bits won't reveal any information about either X or Y, thus it does not increase the external information cost of the protocol.

Since restricting to private randomness does not increase the information cost of any protocol, we still have $\mathrm{IC}^{\mathrm{ext}}_{\mu,\epsilon}(f) = \inf_\pi \mathrm{icost}^{\mathrm{ext}}_{\mu}(\pi)$. □

5.3 Equality VIII

External information complexity can be used to give a simple proof of the $\Omega(n)$ lower bound for the 0-error randomized communication complexity of the equality function.

Theorem 31. *Let μ be the uniform distribution on the set $\{(x,x) : x \in \{0,1\}^n\}$. The 0-error information complexity of* $\textsc{Eq} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *with respect to μ is*

$$\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{Eq}) = n.$$

Proof. $\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{Eq}) = \inf_\pi I\big(X\,Y; \Gamma^{\Pi}_{X,Y}\big) = \inf_\pi H(XY) - H(XY|\Gamma^{\Pi}_{X,Y}\big)$

Now note that for any different pair of (x,x) and (y,y), they must use distinct transcripts if the protocol μ calculates EQUALITY with 0 error probability:

Otherwise (x,x) and (y,y) will be on the same leave of the protocol which implies $(x,y) and (y,x)$ are on the same leave. This contradicts with π's 0 error probability.

Since we have a distinct transcript for each pair of (x,x), we can get the value of XY from the transcript as XY is uniformly distributed over $(x,x) : x \in 0,1^n$. In other words, $H(XY|\Gamma^{\Pi}_{X,Y}\big) = 0$.

Thus $\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{Eq}) = \inf_\pi H(XY) = -\sum \mu(z)\log_2 \mu(z) = -\sum \frac{1}{2^n}\log_2\frac{1}{2^n} = n$ □

5.4 And

Let $\textsc{And} : \{0,1\} \times \{0,1\} \to \{0,1\}$ be the and function $\textsc{And}(x,y) = x \wedge y$ that takes the value 1 if and only if $x = y = 1$.

Theorem 32. *For every distribution μ on $\{0,1\} \times \{0,1\}$,*

$$\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{And}) \le \log_2 3.$$

Furthermore, this bound is tight when $\mu(0,1) = \mu(1,0) = \mu(1,1) = \frac{1}{3}$.

Proof. Think of a protocol where :

-Alice sends Bob her bit, and Bob only send his bit back only if Alice sent 1.
-Alice and Bob calculate the result of AND respectively

It is clear that there are only three possibilities, thus the protocol only has 3 leaves. To represent these leaves, we will only need three different transcripts since there are also no random bits. In other words, for this protocol, $|\Gamma^{\Pi}_{X,Y}| = 3$

Now $\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{And}) = \inf_{\pi} I\big(X\,Y; \Gamma^{\Pi}_{X,Y}\big) \leq H\big(\Gamma^{\Pi}_{X,Y}\big) \leq log_2|\Gamma^{\Pi}_{X,Y}| = log_2 3$.

Furthermore, if $\mu(0,1) = \mu(1,0) = \mu(1,1) = \frac{1}{3}$,

$\mathrm{IC}^{\mathrm{ext}}_{\mu,0}(\textsc{And}) = H(XY)$ by similar from EQUALITY VII

$= -3 \cdot \frac{1}{3} \log_2 \frac{1}{3} = \log_2 3$ □

5.5 Internal information complexity

Another natural notion of information of a protocol is the amount of information that Alice and Bob learn about each other's inputs by running their communication protocol.

Definition 31 (Internal information cost)**.** The *internal information cost* of a randomized protocol Π over the distribution μ on $\mathcal{X} \times \mathcal{Y}$ is

$$\mathrm{icost}^{\mathrm{int}}_{\mu}(\Pi) = I\big(X; \Gamma^{\Pi}_{X,Y}|Y\big) + I\big(Y; \Gamma^{\Pi}_{X,Y}|X\big).$$

where $(X, Y) \sim \mu$.

Definition 32 (Internal information complexity)**.** The *ϵ-error internal information complexity* of the function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with respect to the distribution μ is

$$\mathrm{IC}^{\mathrm{int}}_{\mu,\epsilon}(f) = \inf_{\pi} \mathrm{icost}^{\mathrm{int}}_{\mu}(\pi)$$

with the infimum taken over all 0randomized protocols that compute f with error at most ϵ.

The internal information complexity of a function gives a lower bound on its external information complexity (and therefore on its randomized communication complexity as well).

Theorem 33. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, every distribution μ on $\mathcal{X} \times \mathcal{Y}$, and every $0 \leq \epsilon \leq \frac{1}{2}$,*

$$\mathrm{IC}^{\mathrm{int}}_{\mu,\epsilon}(f) \leq \mathrm{IC}^{\mathrm{ext}}_{\mu,\epsilon}(f).$$

Proof. Let Π be a protocol that computes f with error $\leq \epsilon$ and $\Gamma^{\Pi}_{X,Y} = (M_1 M_2 ... M_k, R)$ without loss of generality.

$\mathrm{IC}^{\mathrm{int}}_{\mu,\epsilon}(f) = I(X; \Gamma^{\Pi}_{X,Y}|Y) + I(Y; \Gamma^{\Pi}_{X,Y}|X)$
$= I(X; M_1 M_2 ... M_k R|Y) + I(Y; M_1 M_2 ... M_k R|Y)$
$= I(X; M_1 M_2 ... M_k|YR) + I(Y; M_1 M_2 ... M_k|YR)$

$I(X; M_1 M_2 ... M_k|YR) = \sum_i I(X; M_i|M_1 M_2 ... M_{i-1} RY)$
$= \sum_{i,sentbyAlice} I(X; M_i|M_1 M_2 ... M_{i-1} RY) + \sum_{i,sentbyBob} I(X; M_i|M_1 M_2 ... M_{i-1} RY)$

$= \sum_{i,sentbyAlice} I(X;M_i|M_1M_2...M_{i-1}RY) + 0$
$\leq \sum_{i,sentbyAlice} I(XY;M_i|M_1M_2...M_{i-1}R)$

Similarly,
$I(Y;M_1M_2...M_k|XR) \leq \sum_{i,sentbyBob} I(XY;M_i|M_1M_2...M_{i-1}R)$

Thus $\mathrm{IC}^{\mathrm{int}}_{\mu,\epsilon}(f) \leq \sum_i I(XY;M_i|M_1M_2...M_{i-1}R) = I(XY;M_1M_2...M_kR) = I(XY;\Gamma^{\Pi}_{X,Y}) = \mathrm{IC}^{\mathrm{ext}}_{\mu,\epsilon}(f)$ □

5.6 Equality VIII

The internal information complexity of a function can be much smaller than its external information complexity, as the following example shows.

Theorem 34. *For every distribution μ over $\{0,1\}^n \times \{0,1\}^n$, the 0-error information complexity of* $\mathrm{EQ} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *is*

$$\mathrm{IC}^{\mathrm{int}}_{\mu,0}(\mathrm{EQ}) = O(1).$$

Proof. Consider the following protocol:

-Alice and Bob chose n linearly independent vectors $r_1, \ldots, r_n \in \{0,1\}^n$ publicly.
-Alice send Bob a bit which is the inner product of r_i and X.
-Bob calculates the inner product of r_i and Y, compare the bit and return the result to Alice.
-If Bob sent 0, both Alice and Bob terminate. Otherwise Alice sends next inner product. If all n inner products were sent, Alice and Bob return 1.

Since all n vectors are linearly independent, they form a basis and this gives us 0 error probability.

Now we calculate the internal information complexity by two cases:

1. if $X = Y$, then both $I\big(X;\Gamma^{\Pi}_{X,Y}|Y\big)$ and $I\big(Y;\Gamma^{\Pi}_{X,Y}|X\big)$ gives a value of 0 since X and Y decides each other if we know $X = Y$ 2. if $X \neq Y$, then it is unlikely that Alice and Bob exchange too many bits:

Since $X \neq Y$, for at least one r_i we'll have different inner products.

The probability of going into ith round is $\frac{2^{n-i+1}-1}{2^n-1} \leq 2^{-i+1}$. So the expected number of rounds will be less than $\sum_{i=1}^{n} 2^{-i+1} = O(1)$. And since we exchange 2 bits each round, the exchanged bits are at most $O(1)$. So $\mathrm{IC}^{\mathrm{int}}_{\mu,0}(\mathrm{EQ}) = O(1)$. □

Hint. Consider a protocol where Alice and Bob use their public randomness to select n linearly independent vectors $r_1, \ldots, r_n \in \{0,1\}^n$.

5.7 Direct sum

Given a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, define $f^k : \mathcal{X}^k \times \mathcal{Y}^k \to \{0,1\}^n$ to be the function where

$$f^n(x^{(1)}, \dots, x^{(k)}, y^{(1)}, \dots, y^{(k)}) = \big(f(x^{(1)}, y^{(1)}), \dots f(x^{(k)}, y^{(k)})\big).$$

This corresponds to the setting where Alice and Bob must compute the value of f on n different pairs of inputs instead of just 1.

It might seem obvious that the complexity of computing f^k is always k times the complexity of computing f, but this is false in general! One counter-example is given by the equality function. Since $R_\epsilon(\textsc{Eq}) = O(\log \frac{1}{\epsilon})$, we have that $R_{2^{-\sqrt{k}}}(\textsc{Eq}) = \sqrt{k}$ and so we might expect that $R_{2^{-\sqrt{k}}}(\textsc{Eq}^k) = k^{3/2}$. In fact, the true randomized complexity of this function is much smaller.

Theorem 35. $R_{2^{-\sqrt{k}}}(\textsc{Eq}^k) = O(k)$.

Proof. Enter your proof here, if you complete it. □

5.8 Direct sum for information complexity

One of the main advantages of working with internal information complexity is that in this setting every function *does* satisfy the intuition that computing k copies of a function is exactly k times harder than computing a single copy of it.

Theorem 36. *For every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ and every $k \geq 1$,*

$$\mathrm{IC}^{\mathrm{int}}_{\mu,0}(f^k) = k \cdot \mathrm{IC}^{\mathrm{int}}_{\mu,0}(f).$$

Proof. Prove inequalities in both directions:

($\leq$) Simply repeat the function f for k times.

($\geq$) Say we have a protocol Π for f^k, we need to design a protocol Π' for f.

$Pub: \;\; X_1 ... X_{i-1}, Y_{i+1} ... Y_k$
$Priv: \;\; X_{i+1} ... X_k, Y_1 ... Y_{i-1}$

$I(Y; \Gamma^{\Pi'} | X) \leq I(Y_i; \Gamma^{\Pi}_{X_{i+1} ... X_k} | i, X_{[i-1]}, X, Y_{i+1}, ..., Y_k)$
$= I(Y_i; \Gamma^{\Pi} | i, X_{[k]}, Y_{i+1}, ..., Y_k)$
$= \frac{1}{k} \sum_{j=1}^{k} I(Y_i; \Gamma^{\Pi} | [i = j], X_{[k]}, Y_{i+1}, ..., Y_k)$
$= \frac{1}{k} \sum_{j=1}^{k} I(Y_j; \Gamma^{\Pi} | X_{[k]}, Y_{j+1}, ..., Y_k)$
$= \frac{1}{k} I(Y; \Gamma^{\Pi} | X)$
$= \frac{1}{k} \mathrm{IC}^{\mathrm{int}}_{\mu,0}(f^k)$

□

5.9 Set disjointness III

It is possible to obtain tight bounds on the internal information complexity of the And function. These tight bounds have been used to obtain *exact* bounds on the communication complexity of the set disjointness function. To establish the tight asymptotic bound on its randomized communication complexity, however, all we need is the following result.

Lemma 9. *Let μ be the distribution on $\{0,1\} \times \{0,1\}$ defined by $\mu(0,0) = \mu(0,1) = \mu(1,0) = \frac{1}{3}$. For every $\epsilon > 0$, there is a constant $c_\epsilon > 0$ for which*

$$\mathrm{IC}^{\mathrm{int}}_{\mu,\epsilon}(\textsc{And}) = c_\epsilon.$$

This result and (extensions of) the direct sum property of internal information complexity can be used to bound the randomized communication complexity of the set disjointness function.

Theorem 37. *For every (small enough) $\epsilon \geq 0$,*

$$\mathbb{R}_\epsilon(\textsc{Disj}) = \Omega(n).$$

Proof. Solve the DISJ problem using previous theorem:

Say we have a protocol Π that solves DISJ, then we can solve AND problem with it as well since DISJ can be reduced to n copies of AND problem.

First say we have i selected from $[n]$ uniformly at random, and $x_1 \ldots x_{i-1}$, $y_{i+1} \ldots y_k$ selected from μ.

$\mathrm{IC}^{\mathrm{int}}(\textsc{And}) = \mathrm{IC}^{\mathrm{int}}(\textsc{Disj})/n = c_\epsilon$
$\mathrm{IC}^{\mathrm{int}}(\textsc{Disj}) = nc_\epsilon$
Then $R_\epsilon(\textsc{Disj}) \geq \mathrm{IC}^{\mathrm{int}}(\textsc{Disj}) = \Omega(n)$ □

Chapter 6

One-way communication and simultaneous message passing

Every communication complexity question we have studied so far in this course have been set in the model where there are two players, Alice and Bob, who exchange bits with each other to compute functions on their joint inputs. This is of course not the only model of communication that is of interest—other models where we change the number of parties or restrict how messages can be sent between them have also been studied extensively. In this chapter, we examine the communication complexity of functions when the communication between Alice and Bob is restricted.

6.1 One-way communication complexity

The first modification to the standard communication complexity setting that we will consider is the *one-way communication* model, where the only communication taking place during a protocol is from Alice to Bob; then Bob outputs the value of the function.

Definition 33 (One-way protocol)**.** A communication protocol π is a *one-way protocol* if all the internal nodes except the ones right above the leaves in the rooted binary tree $T(\pi)$ are labelled with A.

As in the standard two-way communication setting, we can consider deterministic, public-coin, and private-coin protocols.

Definition 34 (One-way communication complexity)**.** The deterministic and (public-coin) randomized *one-way communication complexities* of a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, denoted

$$D^{\to}(f) \qquad \text{and} \qquad R^{\to}(f)$$

respectively, are the minimum cost of a deterministic one-way protocol that computes f and of a public-coin randomized one-way protocol that computes f with error at most $\frac{1}{3}$, respectively.

Remark 5. We can also define one-way analogues of the other models of communication such as distributional complexity, private-coin randomized complexity, etc.

The communication complexity of some functions remains identical in the one-way and two-way settings.

Theorem 38. *The one-way communication complexity of* $\textsc{Eq} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *satisfies*

$$D^{\to}(\textsc{Eq}) = D(\textsc{Eq}) = \Theta(n) \qquad \textit{and} \qquad R^{\to}(\textsc{Eq}) = R(\textsc{Eq}) = \Theta(1).$$

Proof. i)Since the lower bound of two-way communication protocol is lower than or equals to the lower bound of one-way communication protocol, we have $D^{\to}(\textsc{Eq}) \geq (\textsc{Eq})$, $D^{\to}(\textsc{Eq}) = \Omega(n)$.

Now consider the deterministic protocol where Alice sends Bob the whole message and Bob returns the result of $EQ(x, y)$. The communication cost of this protocol is $n + 1$, thus $D^{\to}(\textsc{Eq}) = O(n)$, which gives $D^{\to}(\textsc{Eq}) = \Theta(n)$.

ii) First note that the public-coin randomized EQ protocol that uses 3 bits of communication with error probability 1/4 is a one-way communication protocol. Thus the upper bound of $R^{\to}(\textsc{Eq})$ is still O(1). Thus $R^{\to}(\textsc{Eq}) = \Theta(1) = R(\textsc{Eq})$. □

6.2 Index function

In general, however, the one-way and two-way complexity of a function can differ greatly. A simple example that highlights this difference is the *index function* $\textsc{Index} : \{0,1\}^n \times [n] \to \{0,1\}$ defined by

$$\textsc{Index}(x, i) = x_i.$$

Theorem 39. *The communication complexity of the* $\textsc{Index} : \{0,1\}^n \times [n] \to \{0,1\}$ *function satisfies*

$$D(\textsc{Index}) = O(\log n) \qquad \textit{and} \qquad D^{\to}(\textsc{Index}) = \Omega(n).$$

Proof. Think of a deterministic protocol where Bob sends the index i in binary to Alice and Alice returns the ith entry to Bob which is a single bit. This gives a upper bound of $O(logn)$ to $D(\textsc{Index})$.

For one-way lower bound, note that if $D^{\to}(\textsc{Index}) \leq n$, there exists some $x \neq x'$ in the same node of the protocol tree.

In other words, $f(x) = f(x') for some x \neq x'$. Then $INDEX(x,i) = INDEX(x',i)$ but this is a contradiction. So we have that $D^{\to}(\textsc{Index}) = \Omega(n)$.

□

6.3 One-way vs. two-way communication complexity

The gap between the one-way and two-way communication complexity of the index function is the largest possible.

Theorem 40. *For every function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$,

$$D^{\to}(f) \leq 2^{D(f)} + 1.$$

Proof. Assume there is a two-way deterministic communication protocol that computes f with cost $D(f)$.

Now we try to simulate this protocol with an one-way communication protocol:

Whenever Alice is waiting for a bit from Bob, she will respond to both cases where Bob sent 0 or 1. Since in original protocol Bob sent Alice at most n bits, now Alice will respond with at most 2^n bits. Plus the bit that Bob may return, we will have that $D^{\to}(f) \leq 2^{D(f)} + 1$. □

6.4 Index II

We can strengthen our lower bound on the one-way communication complexity of the index function to show that it holds in the public-coin randomness model as well.

Theorem 41. *The one-way randomized communication complexity of the* $\textsc{Index} : \{0,1\}^n \times [n] \to \{0,1\}$ *function is*

$$R^{\to}(\textsc{Index}) = \Omega(n).$$

Proof. (Method1: Combintorics)
Consider any one-way randomized communication protocol π that computes INDEX as following:

Alice sends out some bits, say a, that depends on her own private randomness and x, to Bob. Then Bob returns either 0 or 1 according to the received value and i. However, since random protocols are just distribution of deterministic protocols, we can just prove the lower bound for every deterministic version of π, in other words, for any fixed randomness.

Now think Alice's action as a function A acts on x and Bob returns either 0 or 1 according to the value of $A(x)$ and i, we can assume Bob can interpret A(x) as a vector and simply returns the ith entry.

This yields the error probability as the hamming distance between x and $A(x)$ over n, $\frac{HD(x,A(x))}{n}$, for fixed x and Alice's private randomness.

Now to prove a lower bound of $O(n)$, assume $A(x)$ has size cn for some constant c. Now only consider Alice's input with error probability $\leq \frac{1}{4}$, by using Stirling's approximation on the expression $1 + \binom{n}{1} + \binom{n}{1} + ... + \binom{n}{\frac{n}{4}}$ which is for the inputs with error probability $\geq \frac{1}{4}$, we conclude that more than $\frac{1}{2}$ of the inputs are with error probability larger than $\frac{1}{4}$. This gives a lower bound for total error probability of $\frac{1}{8}$, and the lower bound on error probability with cn bits of communication also gives the lower bound $\Omega(n)$ for INDEX.

(Method2: Information Cost)
$|M| \geq H(M) \geq I(X;M) = \sum_{i=1}^{n} I(X_i; M|X_1, ..., X_{i-1})$
$= \sum_{i=1}^{n}[H(X_i|X1, ..., X_{i-1}) - H(X_i|M_1X_1, ..., X_{i-1})] = \sum_{i=1}^{n}[1 - H(X_i|M_1X_1, ..., X_{i-1})]$
$= n - \sum_{i=1}^{n} H(X_i|M_1X_1, ..., X_{i-1})$
$\geq n - \sum_{i=1}^{n} H(X_i|M)$

$H(X_i|M) = \sum_{m \in M} H(X_i|M = m)Pr(M = m)$
$= \sum_{m \in M}(\frac{2}{3}log(\frac{3}{2}) + \frac{1}{3}log3)Pr(M = m)$
$\leq \frac{2}{3}$

Thus we have $R^{\rightarrow}(\textsc{Index}) = \Omega(n)$.

Extra Reading: Fano's Inequality

□

6.5 Disjointness

The lower bound on the index function can be used to obtain a simple proof of the $\Omega(n)$ lower bound on the one-way randomized communication complexity of the disjointness function.

Theorem 42. *The one-way randomized communication complexity of the* $\textsc{Disj} : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ *function is*

$$R^{\rightarrow}(\textsc{Disj}) = \Omega(n).$$

Proof. First, by way of contradiction, assume that we can solve DISJOINTNESS with less than cn bits for constant c and large enough n. Then think of the following special case:

Value of y from Bob is a binary string with exactly one 1 on ith index and $n-1$ 0's on other indices. Then $DISJ(x,y) = 0$ if the ith index of Alice's input is 1 as well and 0 otherwise. In other words, Bob just outputs the ith bit of x, which is an equivalent of INDEX function.

Since now we will be able to compute INDEX with less bits than its lower bound, by contradiction, $R^{\rightarrow}(\textsc{Disj}) = \Omega(n)$. □

Remark 6. Your proof should *not* use our previous lower bounds on the two-way randomized communication complexity of the disjointness function; instead, you should be able to use the lower bound on the one-way communication complexity of the index function to get this result.

6.6 Gap Hamming

Define the *Hamming distance* between strings $x, y \in \{0,1\}^n$ to be $\mathrm{d_{Ham}}(x,y) = |\{i \in [n] : x_i \neq y_i\}|$. The *gap Hamming function* $\mathrm{GHD} : \{0,1\}^n \times \{0,1\}^n \to \{0,1,*\}$ is the partial function defined by

$$\mathrm{GHD}(x,y) = \begin{cases} 1 & \text{if } \mathrm{d_{Ham}}(x,y) = \frac{n}{2} \\ 0 & \text{if } |\mathrm{d_{Ham}}(x,y) - \frac{n}{2}| \geq \sqrt{n} \\ * & \text{otherwise.} \end{cases}$$

(We assume throughout this section that n is even.)

Definition 35. A randomized protocol *computes* the partial function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1,*\}$ with error at most ϵ if for every input $(x,y) \in f^{-1}(0) \cup f^{-1}(1)$, it outputs the value $f(x,y)$ with probability at least $1 - \epsilon$.[1]

Theorem 43. *The one-way randomized communication complexity of the* $\mathrm{GHD} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *function is*

$$R^{\rightarrow}(\mathrm{GHD}) = \Omega(n).$$

Proof. Idea: Use public randomness to reduce: note that running GHD on Alice and Bob assume the public randomness is Alice's input, then if GHP = 0/1 the values from Alice and Bob are negative/positive correlation.

$Pr[a_i = b_i] = Pr[||x_{-i} - r_{-i}|| = \frac{n-1}{2}] \cdot Pr[a_1 = b_1|||x_{-i} - r_{-i}|| = \frac{n-1}{2}] +$
$Pr[||x_{-i} - r_{-i}|| \neq \frac{n-1}{2}] \cdot Pr[a_1 = b_1|||x_{-i} - r_{-i}|| \neq \frac{n-1}{2}]$
$- = O(\frac{1}{\sqrt{n}}) \cdot 1 + (1 - O(\frac{1}{\sqrt{n}})) \cdot \frac{1}{2}$ if $x_i = 1$
$- = O(\frac{1}{\sqrt{n}}) \cdot 0 + (1 - O(\frac{1}{\sqrt{n}})) \cdot \frac{1}{2}$ if $x_i = 0$

Thus $Pr[a_i = b_i] = \frac{1}{2} + \frac{1}{\sqrt{n}}$ if $x = 1$
$Pr[a_i = b_i] = \frac{1}{2} - \frac{1}{\sqrt{n}}$ if $x = 0$

And Alice, Bob are positively/negatively correlated.

□

[1] For the inputs in $f^{-1}(*)$, the protocol is free to output anything.

6.7 Simultaneous message passing

In the *simultaneous message passing (SMP)* model of communication, Alice and Bob both send messages to a third party that we will call the Referee, without seeing each other's transmissions. (We picture both Alice and Bob sending their communications to the referee simultaneously.) Note that the Referee does not see either Alice or Bob's inputs, and must then output the result of the protocol using only the messages received from both parties. In the private-coin model of SMP communication, Alice and Bob each have a private source of randomness that is not visible to any other party.

Definition 36. The *SMP (private randomness) complexity* with error ϵ of a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, denoted

$$R_\epsilon^{\|,\mathrm{priv}}(f),$$

is the minimum (worst-case) communication cost of any private-coin SMP protocol that computes f with error at most ϵ on any input $(x,y) \in \mathcal{X} \times \mathcal{Y}$.

As usual, we write $R^{\|,\mathrm{priv}}(f) = R_{1/3}^{\|,\mathrm{priv}}(f)$.

Theorem 44. *The SMP complexity of every function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ satisfies*

$$R^{\|,\mathrm{priv}}(f) = \Omega(\sqrt{D(f)}).$$

Proof. Idea: Consider the success probability of a protocol in which Alice and Bob send t messages to the referee instead of 1, when all t messages are sent for the same input but with independent randomness.

Let Π be an SMP protocol with cost c, let M_x,V_y be the distance on messages by Alice and Bob on input x and y.

Then $Pr_{m_A \sim M_x, m_B \sim V_y}[R(m_A, m_B) = f(x,y)] \geq \frac{2}{3}$.

Claim: There exists T_x = set of $O(c)$ messages that Alice sends such that for all m_B from Bob,

$|Pr_{m_A \sim T_x}[R(m_A, m_B) = 1] - Pr_{m_A \sim M_x}[R(m_A, m_B) = 1]| \leq \frac{1}{100}$

Note this can be proven with Chernoff Bound and Union Bound.

A similar Claim will be that there exists T_y = set of $O(c)$ messages that Bob sends such that for all m_A from Alice,

$|Pr_{m_B \sim T_y}[R(m_A, m_B) = 1] - Pr_{m_B \sim V_y}[R(m_A, m_B) = 1]| \leq \frac{1}{100}$

Combining both results will give $D(f) \leq O(R^{\|,priv}(f)^2)$

And thus $R^{\|,\mathrm{priv}}(f) = \Omega(\sqrt{D(f)})$. □

6.8 Equality X

Theorem 45. *The SMP complexity of the equality function* $\textsc{Eq} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *is*

$$R^{\|,\mathrm{priv}}(\textsc{Eq}) = \Theta(\sqrt{n}).$$

Proof. By Theorem 7 and the fact that $D(EQ) = O(n)$, we have the lower bound $R^{\|,\mathrm{priv}}(\textsc{Eq}) = \Omega(\sqrt{n})$.

Note: By using (basic but still remarkable) results from error-correcting codes, it is sufficient to design a randomized protocol that computes the partial function $\textsc{Eq}^* : \{0,1\}^n \times \{0,1\}^n \to \{0,1,*\}$ defined by

$$\textsc{Eq}^*(x,y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } \mathrm{d}_{\mathrm{Ham}}(x,y) = \frac{n}{2} \\ * & \text{otherwise} \end{cases}$$

with error at most $\frac{1}{3}$.

□

Chapter 7

Multiparty communication

In this chapter, we move beyond the two-party setting of communication complexity and consider the communication complexity of functions that need to be computed by 3 or more parties.

There are two broad classes of models for multiparty communication complexity. In both of them, the input to the function is divided among the k parties. In the *number in hand* model, each party sees their own input and none of the other players' inputs. This model is a natural extension of our two-party communication model, and it is apparent that it only gets harder when there are more players involved. As a result, all the lower bounds that we have established in previous chapters apply to (natural variants of the functions we have seen for) this setting as well.

In the *number on the forehead* model, by contrast, every player sees *all* the inputs except their own. In this model, adding new players now make the function easier to compute, not harder. The lower bounds we have developed no longer apply, and we need to develop new tools to understand which functions remain hard to compute in this setting.

7.1 Number-on-the-forehead model

In the *k-party number on the forehead* (NOF) model of communication complexity, k players $P_1, \dots, P_k$ aim to compute a function $f : \mathcal{X}^k \to \mathcal{Y}$ for some finite sets $\mathcal{X}$ and $\mathcal{Y}$. On input $x_1, \dots, x_k$, each player P_i sees *all* the inputs *except* x_i. A *protocol* determines which player sends the next bit of communication; the bit sent by player P_i depends on the inputs $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_k$ as well as the preivous messages. All messages communicated by a player are seen by *all* other players. This is known as the *blackboard* setting of communication.

The *cost* of a protocol is the maximum number of bits communicated before the protocol halts, where the maximum is taken over all possible inputs. A protocol *computes* the function f if all the players know $f(x)$ at the end of the protocol. As in the two-player setting, we can represent a protocol as a tree and define the function that it computes as the one determined by the value at each leaf.

The *deterministic communication complexity* of a function $f : \mathcal{X}^k \to \mathcal{Y}$, which will again be denoted by $D(f)$, is the minimum cost of a k-party number-on-the-forehead communication protocol that computes f in the blackboard model.

Theorem 46. *Almost all functions $f : (\{0,1\}^n)^k \to \{0,1\}$ have communication complexity $D(f) = \Omega(n)$.*

Proof. Consider a function that requires all $n \cdot k$ input bits to correctly compute the function. Then for any player to be able to compute this function, he must have all the bits in its input.

For such functions, since a player can see all other $n \cdot (k-1)$ bits, all he requires is any other player to send his own n bits. Then after computation, this player can return the result by simply sending 0 or 1.

This requires at least $n+1$ bits, thus for functions that requires all $n \cdot k$ bits to be correctly computed, its protocols need communication of at least $n+1$ bits, hence we have a lower bound of $\Omega(n)$ on $D(f)$.

(Formula) Note we can count the number of functions with cost c protocol which is
$\leq 2^{2c} \cdot 2^{2^{n(k-1)} \cdot 2^c} \cdot k^{2c}$, and we can get the same result as above.

□

Exercise 23. Show that every function $f : (\{0,1\}^n)^k \to \{0,1\}$ has multiparty communication complexity $D(f) \leq n+1$.

Open Problem 3. Identify any explicit function $f : (\{0,1\}^n)^k \to \{0,1\}$ with $k \geq \log n$ that has communication complexity $\omega(1)$.

7.2 Equality XI

Some functions can be much easier in the multiparty communication complexity model. Consider for instance the extension of the equality function for k players $\textsc{Eq}_k : (\{0,1\}^n)^k \to \{0,1\}$ defined by

$$\textsc{Eq}^k(x_1, \ldots, x_k) = \begin{cases} 1 & \text{if } x_1 = x_2 = \cdots = x_k \\ 0 & \text{otherwise.} \end{cases}$$

Theorem 47. *For any $k \geq 3$, $D(\textsc{Eq}^k) \leq 2$.*

Proof. Consider following multi-party communication protocol:

-Select first and second player, player 1 and player 2
-Player 1 computes the result of $EQ^{k-1}(x_2, x_3, ..., x_k)$
-Player 1 sends the result bit, say r, to the blackboard
-Player 2 computes the result of $EQ^{k-1}(x_1, x_3, ..., x_k)$
-Say the result bit is t, player j then sends $AND(s, t)$ as the output of the protocol.

This protocol correctly computes $\textsc{Eq}^k(x_1, \ldots, x_k)$ as it first checks if $(x_2, ..., x_k)$ are all equal, then checks if $(x_1, x_3, ..., x_k)$ are all equal. If they both are, then all of x_i's for $1 \leq i \leq k$ are equal, and if at least one of them are not all equal, it implies otherwise.

As this protocol correctly computes $\textsc{Eq}^k(x_1, \ldots, x_k)$ and takes 2 bits of communication for any input, we have $D(\textsc{Eq}^k) \leq 2$. □

7.3 Majority Inner Product

Other functions also have very efficient protocols in the multiparty communication complexity setting.

For three bits a, b, c, define the *majority* function $\textsc{Maj} : \{0,1\}^3 \to \{0,1\}$ to be $\textsc{Maj}(a, b, c) = 1$ if $a + b + c \geq 2$ and 0 otherwise. The *Majority inner product* function $\text{MIP} : (\{0,1\}^n)^3 \to \{0,1\}$ is defined by

$$\text{MIP}(x, y, z) = \bigoplus_{i \in [n]} \textsc{Maj}(x_i, y_i, z_i).$$

Theorem 48. $D(\text{MIP}) \leq 3$.

Proof. Consider following multi-party communication protocol:

-Player x first looks at the input of player y and player z
-Player x calculates $\bigoplus_{i \in [n]} Z(y_i, z_i)$ where $Z(y_i, z_i) = 1$ iff $y_i = z_i = 1$
-Player x sends this result to blackboard, say the result is r_x

-Similarly, Player y looks at the input of player x and player z
-Player y calculates $\bigoplus_{i \in [n]} Z(x_i, z_i)$ where $Z(x_i, z_i) = 1$ iff $x_i = z_i = 1$
-Player y sends this result to blackboard, say the result is r_y

-Player z looks at the input of player x and player y
-Player z calculates $\bigoplus_{i \in [n]} Z(x_i, y_i)$ where $Z(x_i, y_i) = 1$ iff $x_i = y_i = 1$
-Say result is r_z, players z then sends $r_x \oplus r_y \oplus r_z$ as the protocol's output

This protocol correctly computes $\mathrm{MIP}(x, y, z)$ by following observation:

-If at i-th position, x,y and z all have one, then this position will increase the value $r_x \oplus r_y \oplus r_z$ by 3, which equals to 1 in binary.
-If at i-th position, two of x,y and z all have one, then this position will also increase the value $r_x \oplus r_y \oplus r_z$ by 1.
-Otherwise, this position does not increase the value $r_x \oplus r_y \oplus r_z$.

This equals the result of $\mathrm{MIP}(x, y, z)$.
As this protocol correctly computes $\mathrm{MIP}(x, y, z)$ and takes 3 bits of communication for any input, we have $D(\mathrm{MIP}) \leq 3$. □

7.4 Generalized Inner Product

Other functions become easier, but non-trivial, when the number of players is greater than 2 but less than $\log n$.

The *generalized inner product* function $\mathrm{GIP}^k : (\{0,1\}^n)^k \to \{0,1\}$ is defined by

$$\mathrm{GIP}^k(x^{(1)}, \ldots, x^{(k)}) = \bigoplus_{i \in [n]} x_i^{(1)} \wedge x_i^{(2)} \wedge \cdots \wedge x_i^{(k)}.$$

When we view the input to the function as a $k \times n$ matrix, the function takes the value 1 if and only if the number of all-one columns of this matrix is odd.

Theorem 49. $D(\mathrm{GIP}^k) = O(\frac{nk}{2^k})$.

Proof. Consider the following protocol:

First we divide the n bits into blocks with size $2^{k-1}-1$, where the last block may have size $\leq 2^{k-1}-1$

Then any player, say player 1, will be able to decide at least one column with size k that does not exist in each block. This is because there are 2^{k-1} combinations for last $k-1$ bits, so player 1 can simply choose the column with size $k-1$ that a block doesn't have, then fill either 0 or 1 at the beginning.

Player 1 sends all determined vectors to all other players. Then for each block, let the vector that doesn't exist in this block be v. We rearrange the order of bits each player holds and bits in v by the same permutation in each block, so that v is in the format $(0, 0, 0, ..., 0, 1, 1, 1, ..., 1, 1)$ where the first 1 is at the l-th position.

On the resulted $\leq 2^{k-1} - 1$ vectors, we do the following:
For $1 \leq i \leq l$

Player i(index after permutation) computes the parity of number of columns in this block with following format:

$(0, 0, 0, ..., 0, *, 1, 1, ..., 1)$ where the $*$ occurs at the i-th position.

Then player i sends this result to all other players.

End

Then all players can sum over the results sent by each player and all get the result of $\mathrm{GIP}^k(x^{(1)}, \ldots, x^{(k)})$.

Proof of correctness:
Let p_i be the parity of the vector $(0,0,0,...,0,1,1,...,1,1)$ where the first 1 is at i-th position. Then at each round, the parity of $(0,0,0,....,0,*,1,1,...,1)$ for player i we calculated is actually $p_i + p_{i+1}$. And if we sum this over, we will get the value of $p_1 + p_l$.
However, by our setting, we know that the vector v does not exist in the current block, thus the value of $p_l = 0$, and the value we get is actually p_1, which is the parity of all-1 columns in current block. Then we get the answer if all players sum all results up.

Run time:
Sends all v vectors: $\frac{n}{2^{k-1}-1} \cdot k$ (# of blocks · length)
Result of each player: $\frac{n}{2^{k-1}-1} \cdot l \leq \frac{n}{2^{k-1}-1} \cdot k$ (# of blocks · iterations)
Thus $D(\mathrm{GIP}^k) = O(\frac{nk}{2^k})$.
Reference: The BNS Lower Bound for MultiParty Protocols is Nearly Optimal, V.Grolmusz □

7.5 Cylinder intersections

Rectangles have a natural generalization in the number-on-the-forehead model in the form of *cylinder intersections.*

Definition 37. A *cylinder* in $\mathcal{X}^k$ *in the* i*th direction* is a set $S \subseteq \mathcal{X}^k$ such that whether an element $(x^{(1)}, \ldots, x^{(k)}) \in S$ or not is independent of the value of $x^{(i)}$.

Definition 38. The set $S \subseteq \mathcal{X}^k$ is a *cylinder intersection* if $S = \cap_{i\in[k]} S_i$ where each S_i is a cylinder in the ith direction.

Cylinder intersections can be used to bound the multiparty communication complexity of functions via the following fundamental lemma.

Lemma 10. *Every multiparty protocol that computes f and has cost c partitions the domain of f into at most 2^c monochromatic cylinder intersections.*

Proof. Under the number of forehead setting, players send they bits according to the input of all other players and the bits that has been communicated. In other words, the bits sent by player i does not depend on the input for player i which is $x^{(i)}$.

This can be viewed as that a bit sent by player i is defined by some cylinders in the i-th direction and the output of the protocol can be defined by the intersection of these cylinders.

Thus if the communication takes c bits, there are at most 2^c monochromatic possible cylinder intersections. □

7.6 Discrepancy

Having generalized the notion of rectangles to the multiparty setting, we could hope that all the lower bound techniques also generalize. That unfortunately does not appear to be the case. The only technique that generalizes to this setting is discrepancy.

Definition 39. For any distribution μ on $\mathcal{X}^k$, the *μ-discrepancy* of $f : \mathcal{X}^k \to \{0,1\}$ is

$$\mathrm{disc}_\mu(f) = \max_S \left|\mu\big(S \cap f^{-1}(0)\big) - \mu\big(S \cap f^{-1}(1)\big)\right|$$

where the maximum is over all cylinder intersections $S \subseteq \mathcal{X}^k$.

Lemma 11. *For every function $f : \mathcal{X}^k \to \{0,1\}$ and distribution μ on $\mathcal{X}^k$,*

$$D(f) = \Omega\left(\log \frac{1}{\mathrm{disc}_\mu(f)}\right).$$

Proof. (Method1)We can use the similar proof for the discrepancy bound for the two-party case.

If the protocol π computes the function f, then we have that:

$Pr[\pi(X) = f(X)] = 1$
$Pr[\pi(X) \neq f(X)] = 0$ where $X \in \mathcal{X}^k$
Thus $Pr[\pi(X) = f(X)] - Pr[\pi(X) \neq f(X)] = 1$

Then we have that for any cylinder intersections partitions, protocol π and distribution μ
$\sum_S \sum_{X \in S} Pr[select\ X]\ *\ (Pr[\pi(X) = f(X)] - Pr[\pi(X) \neq f(X)]) = 1$
$\sum_S \sum_{X \in S} \mu(X)\ *\ |Pr[\pi(X) = f(X)] - Pr[\pi(X) \neq f(X)]| \geq 1$
$\sum_S \sum_{X \in S} \mu(X)\ *\ |Pr[\pi(X) = 0] - Pr[\pi(X) = 1]| \geq 1$
$\sum_S \sum_{X \in S} |\mu(X)Pr[\pi(X) = 0] - \mu(X)Pr[\pi(X) = 1]| \geq 1$
$\sum_S |\mu\big(S \cap f^{-1}(0)\big) - \mu\big(S \cap f^{-1}(1)\big)| \geq 1$
$\alpha * \max_S |\mu\big(S \cap f^{-1}(0)\big) - \mu\big(S \cap f^{-1}(1)\big)| \geq 1$ where α is the greatest possible number of cylinder intersections (monochromatic for this case)
$\alpha * \mathrm{disc}_\mu(f) \geq 1$
$\alpha \geq \frac{1}{\mathrm{disc}_\mu(f)}$

By previous lemma, when there is at least α cylinder intersections, there must be at least $log\alpha$ communication bits, thus $D(f) = \Omega\left(\log \frac{1}{\mathrm{disc}_\mu(f)}\right)$.

(Method2)
Let M be the cylinder with largest $\mu(C_i)$, and say there are l cylinders,

$1 = \mu(C_1 \cup C_2 \cup C_3 ... \cup C_l)$
$\leq \mu(C_1) + \mu(C_2) + ... + \mu(C_l)$
$\leq l\mu(M)$
$\leq 2^d\mu(M)$

Thus $d \geq log\frac{1}{disc_\mu(f)}$ □

Exercise 24. We can also consider *randomized* multiparty communication protocols. Show that for every function f, every distribution μ, and every $\epsilon \leq \frac{1}{2}$, we also have

$$R_\epsilon(f) \geq \log \frac{1-2\epsilon}{\mathrm{disc}_\mu(f)}.$$

7.7 Generalized inner product II

We can use the discrepancy bound to give a lower bound on the communication complexity of the generalized inner product function.

Theorem 50. $D(\mathrm{GIP}^k) = \Omega(\frac{n}{4^k})$.

Proof. We consider the discrepancy of GIP^k under uniform distribution μ and solve it by induction on k.

First think about the base case where $k = 2$, in Chapter 3 we have proven that $\mathrm{disc}_\mu(IP) \leq 2^{-n/2} \leq 2^{-\frac{n}{4^k}}$.

Now we prove that if for $k = i$ the statement $\mathrm{disc}_\mu(GIP^k) \leq 2^{-\frac{n}{4^k}}$ stands true, then for $k = k+1$ the statement is also true by inductive steps.

Say for $k = i$, we have that $\mathrm{disc}_\mu(GIP^i) = \max_S \left|\mu\big(S \cap f^{-1}(0)\big) - \mu\big(S \cap f^{-1}(1)\big)\right| \leq 2^{-\frac{n}{4^i}}$. Then consider the case for $k = i+1$, we are adding a vector of dimension n that is uniformly chosen at random from $\{0,1\}^k$ and compute the new result of GIP^{k+1}.

First note that since our distribution is uniform, the probability of any index in the matrix to be 0 is $\frac{1}{2}$, and if one index in any column is 0, the same position of GIP^k will be 0.
Thus by adding a new vector, we increase the expectation value of $\mu\big(S \cap f^{-1}(0)\big)$ by $\frac{1}{2} * \mu\big(S \cap f^{-1}(1)\big)$ and decrease the value of $\mu\big(S \cap f^{-1}(1)\big)$ by half. Then the value of discrepancy will decrease by a factor of 4, and thus we have for $k = i+1$, $\mathrm{disc}_\mu(GIP^k) \leq 2^{-\frac{n}{4^{i+1}}} = 2^{-\frac{n}{4^k}}$.

Thus we have that $\mathrm{disc}_\mu(GIP^k) \leq 2^{-\frac{n}{4^k}}$, and by the previous lemma we have that
$D(GIP^k) \geq \log \frac{1}{2^{-\frac{n}{4^k}}} = \log 2^{\frac{n}{4^k}} = \Omega(\frac{n}{4^k})$.

Extra Reading:
1. Cauchy-Schwarz inequality and some elementary arguments can prove this theorem as well. 2. Higher order of Fourier Analysis and Gowers uniformly norm.

□

7.8 Multiparty Simultaneous Message Passing

In the *simultaneous message passing* (SMP) model of multiparty communication complexity, we again consider the number-on-the-forehead model with blackboard communication with one change:

all the players now write their messages on the blackboard simultaneously. Or, equivalently, all the bits communicated by the protocol depend on the other players inputs but *not* on the messages previously sent.

The minimum cost of an SMP multiparty communication protocol that computes a function $f : \mathcal{X}^k \to \{0,1\}$ is denoted by $D^{\|}(f)$.

A particularly interesting function to study in the multiparty SMP model is the k-party generalization of the Index function where $\textsc{Index}^k : \{0,1\}^{n^{k-1}} \times [n]^{k-1} \to \{0,1\}$ where player P_1's input is a $(k-1)$-dimensional array of bits, the $k-1$ other players all receive an index in $[n]$, and

$$\textsc{Index}^k(A, i_2, \ldots, i_k) = A[i_2, \ldots, i_k].$$

Theorem 51. *$D^{\|}(\textsc{Index}^k) = \Omega(n/k)$.*

Proof. We can reduce this problem to an one-way two-party index problem. Say we have $\textsc{Index}^k(A, i_2, \ldots, i_k) = A[i_2, \ldots, i_k]$, where Alice holds A and Bob holds $i_2, \ldots, i_k$.

Assume there is an one-way protocol Π that solves this two-party version with $< \frac{n}{k}$ bits.

Now in mutli-party case consider player $j \in 2, ..., k$.

The other indices Player j sees: $[n]^{k-2}$
For each $t \in [n]^{k-2}$,
Alice sends Bob message j from Π on the input (A, t).

This gives us that
$n^{k-2} \cdot k = n^{k-1}$, and thus $D^{\|}(\textsc{Index}^k) = \Omega(n/k)$.

□

Exercise 25. Show that $D(\textsc{Index}^k) = \Theta(\log n)$.

7.9 Sum Index

Another interesting variant of the index function is the 3-party function $\textsc{SumIndex} : \{0,1\}^n \times [n] \times [n]$ defined by

$$\textsc{SumIndex}(A, i, j) = A[i \oplus j]$$

with $i \oplus j$ being the bitwise OR operation. (It is useful to consider only the case where n is a power of 2 for simplicity.)

Theorem 52. *$D^{\|}(\textsc{SumIndex}) = \Omega(\sqrt{n})$.*

Proof. Say there is a protocol Π that solves SUM INDEX problem, we can solve INDEX problem in one-way setting with Π as well.

Say Alice has input $A \in \{0,1\}^n$ which is equivalent to A in SUM INDEX, Say Bob has input $l \in [n]$ which we will convert to (i, j) such that $i \oplus j = l$.

Now Alice simulates player 2+3: communication$= 2 \cdot c \cdot 2^{\frac{logn}{2}} = 2c\sqrt{n} = \Omega(n)$ bits.
Then Bobs receives message from Alice, simulates referee and outputs the result.

Thus we have $c = \Omega(\sqrt{n})$ bits of communication, and by reduction and lower bound of one-way INDEX problem, we hanve that $D^{\|}(\textsc{SumIndex}) = \Omega(\sqrt{n})$.

□

Exercise 26. Prove that $D^{\|}(\textsc{SumIndex}) = o(n)$.

Open Problem 4. The best bounds on the SMP complexity of the SumIndex function are

$$\Omega(\sqrt{n}) \leq D^{\|}(\textsc{SumIndex}) \leq O(n^{0.73}).$$

Can you improve either the upper or the lower bound?