Computational Complexity: A Modern Approach

Draft of a book: Dated January 2007

`Comments welcome!`

Sanjeev Arora and Boaz Barak
Princeton University
complexitybook@gmail.com

This is an Internet draft. Some chapters are more finished than others. References and attributions are very preliminary and we apologize in advance for any omissions (but hope you will nevertheless point them out to us).

Please send us bugs, typos, missing references or general comments to complexitybook@gmail.com — **Thank You!!**

Chapter 12

Communication Complexity

Communication complexity concerns the following scenario. There are two players with unlimited computational power, each of whom holds an n bit input, say x and y. Neither knows the other's input, and they wish to collaboratively compute $f(x, y)$ where function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is known to both. Furthermore, they had foreseen this situation (e.g., one of the parties could be a spacecraft and the other could be the base station on earth), so they had already —before they knew their inputs x, y— agreed upon a protocol for communication[1]. The *cost* of this protocol is the *number of bits communicated* by the players for the *worst-case* choice of x, y.

Researchers have studied many modifications of the above basic scenario, including randomized protocols, nondeterministic protocols, average-case protocols (where x, y are assumed to come from a distribution), multiparty protocols, etc. Truly, this is a self-contained mini-world within complexity theory. Furthermore, lowerbounds on communication complexity have uses in a variety of areas, including lowerbounds for parallel and VLSI computation, circuit lowerbounds, polyhedral theory, data structure lowerbounds, etc. We give a very rudimentary introduction to this area; an excellent and detailed treatment can be found in the book by Kushilevitz and Nisan [**?**].

12.1 Definition

Now we formalize the informal description of communication complexity given above.

A t-round *communication protocol* for f is a sequence of function pairs $(S_1, C_1), (S_2, C_2), \ldots, (S_t, C_t), (f_1, f_2)$. The input of S_i is the communication pattern of the first $i-1$ rounds and the output is from $\{1, 2\}$, indicating which player will communicate in the ith round. The input of C_i is the input string of this selected player as well as the communication pattern of the first $i-1$ rounds. The output of C_i is the bit that this player will communicate in the ith round. Finally, f_1, f_2 are 0/1-valued functions that the players apply at the end of the protocol to their inputs as well as the communication pattern in the t rounds in order to compute the output. These two outputs must be $f(x, y)$. The

[1] Do not confuse this situation with *information theory*, where an algorithm is given messages that have to be transmitted over a noisy channel, and the goal is to transmit them robustly while minimizing the amount of communication. In communication complexity the channel is not noisy and the players determine what messages to send.

communication complexity of f is

$$C(f) = \min_{\text{protocols } \mathcal{P}} \max_{x,y} \quad \{\text{Number of bits exchanged by } \mathcal{P} \text{ on } x, y.\}$$

Notice, $C(f) \leq n+1$ since the trivial protocol is for one player to communicate his entire input, whereupon the second player computes $f(x, y)$ and communicates that single bit to the first. Can they manage with less communication?

Example 12.1 (Parity)
Suppose the function $f(x, y)$ is the *parity* of all the bits in x, y. We claim that $C(f) = 2$. Clearly, $C(f) \geq 2$ since the function depends nontrivially on each input, so each player must transmit at least one bit. Next, $C(f) \leq 2$ since it suffices for each player to transmit the parity of all the bits in his possession; then both know the parity of all the bits.

Remark 12.2
Sometimes students ask whether a player can communicate by not saying anything? (After all, they have three options: send a 0, or 1, or not say anything in that round.) We can regard such protocols as communicating with a ternary, not binary, alphabet, and analyze them analogously.

12.2 Lowerbound methods

Now we discuss methods for proving lowerbounds on communication complexity. As a running example in this chapter, we will use the equality function:

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

We will see that $C(EQ) \geq n$.

12.2.1 Fooling set

We show $C(EQ) \geq n$. For contradiction's sake, suppose a protocol exists whose complexity is at most $n - 1$. Then there are only 2^{n-1} communication patterns possible between the players. Consider the set of all 2^n pairs (x, x). Using the pigeonhole principle we conclude there exist two pairs (x, x) and (x', x') on which the communication pattern is the same. Of course, thus far we have nothing to object to, since the answers $\mathrm{EQ}(x, x)$ and $\mathrm{EQ}(x', x')$ on both pairs are 1. However, now imagine giving one player x and the other player x' as inputs. A moment's thought shows that the communication pattern will be the same as the one on (x, x) and (x', x'). (Formally, this can be shown by induction. If player 1 communicates a bit in the first round, then clearly this bit is the same whether his input is x or x'. If player 2 communicates in the 2nd round, then his bit must also be the same on both inputs since he receives the same bit from player 1. And so on.)

Hence the player's answer on (x, x) must agree with their answer on (x, x'). But then the protocol must be incorrect, since $\mathrm{EQ}(x, x') = 0 \neq \mathrm{EQ}(x, x)$.

The lowerbound argument above is called a *fooling set* argument. It is formalized as follows.

DEFINITION 12.3
A *fooling set* for $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is a set $S \subseteq \{0,1\}^n \times \{0,1\}^n$ and a value $b \in \{0,1\}$ such that:

1. For every $(x, y) \in S$, $f(x, y) = b$.
2. For every two distinct pairs $(x_1, y_1), (x_2, y_2) \in S$, either $f(x_1, y_2) \neq b$ or $f(x_2, y_1) \neq b$.

LEMMA 12.4
If f has a fooling set with m pairs then $C(f) \geq \log_2 m$.

EXAMPLE 12.5 (DISJOINTNESS)
Let x, y be interpreted as characteristic vectors of subsets of $\{1, 2, \ldots, n\}$. Let $\mathrm{DISJ}(x, y) = 1$ if these two subsets are disjoint, otherwise $\mathrm{DISJ}(x, y) = 0$. Then $C(\mathrm{DISJ}) \geq n$ since the following 2^n pairs constitute a fooling set:

$$S = \left\{(A, \overline{A}) : A \subseteq \{1, 2, \ldots, n\}\right\}.$$

12.2.2 The tiling lowerbound

The tiling lowerbound takes a more global view of f. Consider the matrix of f, denoted $M(f)$, which is a $2^n \times 2^n$ matrix whose (x, y)'th entry is $f(x, y)$. See Figure 12.1. We visualize the communication protocol in terms of this matrix. A *combinatorial rectangle* (or just rectangle) in the matrix is a submatrix corresponding to $A \times B$ where $A \subseteq \{0,1\}^n$, $B \subseteq \{0,1\}^n$. If the protocol begins with the first player sending a bit, then $M(f)$ partitions into two rectangles of the type $A_0 \times \{0,1\}^n$, $A_1 \times B^n$, where A_b is the subset of strings for which the first player communicates bit b. Notice, $A_0 \cup A_1 = \{0,1\}^n$. If the next bit is sent by the second player, then each of the two rectangles above is further partitioned into two smaller rectangles depending upon what this bit was. If the protocol continues for k steps, the matrix gets partitioned into 2^k rectangles. Note that each rectangle in the partition corresponds to a subset of input pairs for which the communication sequence thus far has been identical. (See Figure 12.2 for an example.)

Figure unavailable in pdf file.

Figure 12.1: Matrix M(f) for the equality function when the inputs to the players have 3 bits. The numbers in the matrix are values of f.

Figure unavailable in pdf file.

Figure 12.2: Two-way communication matrix after two steps. The large number labels are the concatenation of the bit sent by the first player with the bit sent by the second player.

If the protocol stops, then the value of f is determined within each rectangle, and thus must be the same for all pairs x, y in that rectangle. Thus the set of all communication patterns must lead to a partition of the matrix into *monochromatic* rectangles. (A rectangle $A \times B$ is *monochromatic* if for all x in A and y in B, $f(x, y)$ is the same.)

Definition 12.6

A monochromatic tiling of $M(f)$ is a partition of $M(f)$ into disjoint monochromatic rectangles. We denote by $\chi(f)$ the minimum number of rectangles in any monochromatic tiling of $M(f)$.

The following theorem is immediate from our discussion above.

Theorem 12.7

If f has communication complexity C then it has a monochromatic tiling with at most 2^C rectangles. Consequently, $C \geq \log_2 \chi(f)$.

The following observation shows that the tiling bound subsumes the fooling set bound.

Lemma 12.8

If f has a fooling set with m pairs, then $\chi(f) \geq m$.

Proof: If (x_1, y_1) and (x_2, y_2) are two of the pairs in the fooling set, then they cannot be in a monochromatic rectangle since not all of $(x_1, y_1), (x_2, y_2), (x_1, y_2), (x_2, y_1)$ have the same f value. ■

12.2.3 Rank lowerbound

Now we introduce an algebraic method to lowerbound $\chi(f)$ (and hence communication complexity). Recall the high school notion of *rank* of a square matrix: it is the size of the largest subset of rows/colums that are independent. The following is another definition.

Definition 12.9

If a matrix has entries from a field F then the *rank* of an $n \times n$ matrix M is the minimum value of l such that M can be expressed as

$$M = \sum_{i=1}^{l} \alpha_i B_i,$$

where $\alpha_i \in F \setminus \{0\}$ and each B_i is an $n \times n$ matrix of rank 1.

Note that $0, 1$ are elements of every field, so we can compute the rank over any field we like. The choice of field can be crucial; see Problem 5 in the exercises.

The following theorem is trivial, since each monochromatic rectangle can be viewed (by filling out entries outside the rectangle with 0's) as a matrix of rank at most 1 .

Theorem 12.10

For every function f, $\chi(f) \geq$ rank$(M(f))$.

12.2.4 Discrepancy

The *discrepancy* of a rectangle $A \times B$ in $M(f)$ is

$$\frac{1}{2^{2n}} \left| \text{number of 1's in } A \times B - \text{number of 0's in } A \times B \right|. \tag{1}$$

The *discrepancy* of the matrix $M(f)$, denote $\mathrm{Disc}(f)$, is the largest discrepancy among all rectangles. The following Lemma relates it to $\chi(f)$.

Lemma 12.11

$$\chi(f) \geq \frac{1}{Disc(f)}.$$

Proof: For a monochromatic rectangle, the discrepancy is its size divided by 2^{2n}. The total number of entries in the matrix is 2^{2n}. The bound follows. ■

Example 12.12
Lemma 12.11 can be very loose. For the $EQ()$ function, the discrepancy is at least $1-2^{-n}$ (namely, the discrepancy of the entire matrix), which would only give a lowerbound of 2 for $\chi(f)$. However, $\chi(f)$ is at least 2^n, as already noted.

Now we describe a method to upperbound the discrepancy using *eigenvalues*.

Lemma 12.13 (eigenvalue bound)
For any matrix M, the discrepancy of a rectangle $A \times B$ is at most $\lambda_{max}(M)\sqrt{|A|\,|B|}/2^{2n}$, where $\lambda_{max}(M)$ is the magnitude of the largest eigenvalue of M.

Proof: Let $1_A, 1_B \in \mathbb{R}^n$ denote the characteristic vectors of A, B. Then $|1_A|_2 = \sqrt{\sum_{i \in A} 1^2} = \sqrt{|A|}$.

The discrepancy of the rectangle $A \times B$ is

$$\frac{1}{2^{2n}} 1_A^T M 1_B \leq \frac{1}{2^{2n}} \lambda_{max}(M) \left| 1_A^T 1_B \right| \leq \frac{1}{2^{2n}} \lambda_{max}(M) \sqrt{|A|\,|B|}.$$

explain this.

■

Example 12.14
The *mod 2 inner product* function defined as $f(x,y) = (x \cdot y)_2 = \sum_i x_i y_i (\text{mod} 2)$ has been encountered a few times in this book. To bound its discrepancy, we consider the matrix $2M(f) - 1$. This transformation makes the range of the function $\{-1, 1\}$ and will be useful again later. Let this new

matrix be denoted N. It is easily checked that every two distinct rows (columns) of N are orthogonal, every row has ℓ_2 norm $2^{n/2}$, and that $N^T = N$. Thus we conclude that $N^2 = 2^n I$ where I is the unit matrix. Hence every eigenvalue is either $+2^{n/2}$ or $-2^{n/2}$, and thus Lemma 12.13 implies that the discrepancy of a rectangle $A \times B$ is at most $2^{n/2}\sqrt{|A|\,|B|}$ and the overall discrepancy is at most $2^{3n/2}$ (since $|A|\,, |B| \leq 2^n$).

A technique for upperbounding the discrepancy

Now we describe an upperbound technique for the discrepancy that will later be useful in the multiparty setting (Section 12.3). For ease of notation, in this section we change the range of f to $\{-1, 1\}$ by replacing 1's in $M(f)$ with -1's and replacing 0's with 1's. Note that now

$$\mathrm{Disc}(f) = \max_{A,B} \frac{1}{2^{2n}} \left| \sum_{a \in A, b \in B} f(a,b) \right| .$$

Definition 12.15
$\mathcal{E}(f) = \mathbf{E}_{a_1,a_2,b_1,b_2}\left[\prod_{i=1,2}\prod_{j=1,2} f(a_i, b_j)\right]$.

Note that $\mathcal{E}(f)$ can be computed, like the rank, in polynomial time given the $M(f)$ as input.

Lemma 12.16

$$\mathit{Disc}(f) \leq \mathcal{E}(f)^{1/4}.$$

Proof: The proof follows in two steps.

Claim 1: *For every function* $h : \{0,1\}^n \times \{0,1\}^n \to \{1, -1\}$, $\mathcal{E}(h) \geq \left(\mathbf{E}_{a,b}[f(a,b)]\right)^4$.

We will use the Cauchy-Schwartz inequality, specifically, the version according to which $\mathbf{E}[z^2] \geq (\mathbf{E}[z])^2$ for every random variable z.

$$\begin{aligned}
\mathcal{E}(h) &= \mathbf{E}_{a_1,a_2}\left[\mathbf{E}_{b_1,b_2}\left[\prod_{i=1,2}\prod_{j=1,2} h(a_i,b_j)\right]\right] && (2)\\
&= \mathbf{E}_{a_1,a_2}\left[\left(\mathbf{E}_b[h(a_1,b)h(a_2,b)]\right)^2\right] && (3)\\
&\geq \left(\mathbf{E}_{a_1,a_2}\left[\mathbf{E}_b[h(a_1,b)h(a_2,b)]\right]\right)^2 \qquad \text{(Cauchy Schwartz)} && (4)\\
&\geq \left(\mathbf{E}_{a,b}[h(a,b)]\right)^4 . \qquad \text{(repeating prev. two steps)} && (5)
\end{aligned}$$

Claim 2: *For every function* f *there is a function* h *such that* $\mathcal{E}(f) = \mathcal{E}(h)$ *and* $\mathbf{E}_{a,b}[h(a,b)] \geq$ *Disc*(f).

First, we note that for every two functions $g_1, g_2 : \{0,1\}^n \to \{-1,1\}$, if we define $h = f \circ g_1 \circ g_2$ as

$$h(a,b) = f(a,b)g_1(a)g_2(b)$$

then $\mathcal{E}(f) = \mathcal{E}(h)$. The reason is that for all a_1, a_2, b_1, b_2,

$$\prod_{i=1,2}\prod_{j=1,1} h(a_i,b_j) = g_1(a_1)^2 g_1(a_2)^2 g_2(b_1)^2 g_2(b_2)^2 \prod_{i=1,2}\prod_{j=1,2} f(a_i,b_j)$$

and the square of any value of g_1, g_2 is 1.

Now we prove Claim 2 using the probabilistic method. Define two random functions $g_1, g_2 : \{0,1\}^n \to \{-1,1\}$ as follows:

$$\begin{aligned} g_1(a) &= \begin{cases} 1 & \text{if } a \in A \\ r_a & r_a \in \{-1,1\} \text{ is randomly chosen} \end{cases} \\ g_2(b) &= \begin{cases} 1 & \text{if } b \in B \\ s_b & s_b \in \{-1,1\} \text{ is randomly chosen} \end{cases} \end{aligned}$$

Let $h = f \circ g_1 \circ g_2$, and therefore $\mathcal{E}(h) = \mathcal{E}(f)$. Furthermore

$$\begin{aligned} \mathbf{E}_{g_1,g_2}\left[\mathbf{E}_{a,b}[h(a,b)]\right] &= \mathbf{E}_{a,b}\left[\mathbf{E}_{g_1,g_2}[f(a,b)g_1(a)g_2(b)]\right] && (6) \\ &= \frac{1}{2^{2n}} \sum_{a\in A, b\in B} f(a,b) && (7) \\ &= \mathrm{Disc}(f) && (8) \end{aligned}$$

where the second line follows from the fact that $\mathbf{E}_{g_1}[g_1(a)] = \mathbf{E}_{g_2}[g_2(b)] = 0$ for $a \notin A$ and $b \notin B$.

Thus in particular there *exist* g_1, g_2 such that $|\mathbf{E}_{a,b}[h(a,b)]| \geq \mathrm{Disc}(f)$. ■

12.2.5 Comparison of the lowerbound methods

As already noted, discrepancy upperbounds imply lowerbounds on $\chi(f)$. Of the other three methods, the tiling argument is the strongest, since it subsumes the other two. The rank method is the weakest, since the rank lowerbound always implies a tiling lowerbound and a fooling set lowerbound (the latter follows from Problem 3 in the exercises).

Also, we can separate the power of these lowerbound arguments. For instance, we know functions for which there is a significant gap between $\log \chi(f)$ and $\log \mathrm{rank}(M(f))$. However, the following conjecture (we only state one form of it) says that all three methods (except discrepancy, which as already noted can be arbitrarily far from $\chi(f)$) give the same bound up to a polynomial factor.

Conjecture 12.17 (log rank conjecture)
There is a constant $c > 1$ such that $C(f) = O(\log(rank(M(f)))^c)$ for all f and all input sizes n.

12.3 Multiparty communication complexity

There is more than one way to generalize communication complexity to a multiplayer setting. The most interesting model is the "number on the forehead" model often encountered in math puzzles that involve people in a room, each person having a bit on their head which everybody else can see but they cannot. More formally, there is some function $f : (\{0,1\}^n)^k \to \{0,1\}$, and the input is $(x_1, x_2, \ldots, x_k)$ where each $x_i \in \{0,1\}^n$. The ith player can see all the x_j such that $j \neq i$. As in the 2-player case, the k players have an agreed-upon protocol for communication, and all this communication is posted on a "public blackboard". At the end of the protocol all parties must know $f(x_1, \ldots, x_k)$.

Example 12.18
Consider computing the function

$$f(x_1, x_2, x_3) = \bigoplus_{i=1}^{n} \mathrm{maj}(x_{1i}, x_{2i}, x_{3i})$$

in the 3-party model where x_1, x_2, x_3 are n bit strings. The communication complexity of this function is 3: each player counts the number of i's such that she can determine the majority of x_{1i}, x_{2i}, x_{3i} by examining the bits available to her. She writes the parity of this number on the blackboard, and the final answer is the parity of the players' bits. This protocol is correct because the majority for each row is known by either 1 or 3 players, and both are odd numbers.

Example 12.19 (Generalized Inner Product)
The *generalized inner product function $GIP_{k,n}$* maps nk bits to 1 bit as follows

$$f(x_1, \ldots, x_k) = \bigoplus_{i=1}^{n} \bigwedge_{j=1}^{k} x_{ij}. \tag{9}$$

Notice, for $k = 2$ this reduces to the mod 2 inner product of Example 12.14.

In the 2-party model we introduced the notion of a monochromatic rectangle in order to prove lower bounds. For the k-party case we will use cylinder intersections. A *cylinder in dimension i* is a subset S of the inputs such that if $(x_1, \ldots, x_k) \in S$ then for all x_i' we have that $(x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_k) \in S$ also. A *cylinder intersection* is $\cap_{i=1}^{k} T_i$ where T_i is a cylinder in dimension i.

As noted in the 2-party case, a communication protocol can be viewed as a way of partitioning the matrix $M(f)$. Here $M(f)$ is a k-dimensional cube, and player i's communication does not depend upon x_i. Thus we conclude that if f has a multiparty protocol that communicates c bits, then its matrix has a tiling using at most 2^c monochromatic cylinder intersections.

Lemma 12.20
If every partition of $M(f)$ into monochromatic cylinder intersections requires at least R cylinder intersections, then the k-party communication complexity isat least $\log_2 R$.

Discrepancy-based lowerbound

In this section, we will assume as in our earlier discussion of discrepancy that the range of the function f is $\{-1, 1\}$. We define the *k-party discrepancy* of f by analogy to the 2-party case

$$\mathrm{Disc}(f) = \frac{1}{2^{nk}} \max_T \left| \sum_{(a_1,a_2,\ldots,a_k)\in T} f(a_1, a_2, \ldots, a_k) \right|,$$

where T ranges over all cylinder intersections.

To upperbound the discrepancy we introduce the k-party analogue of $\mathcal{E}(f)$. Let a *cube* be a set D in $\{0,1\}^{nk}$ of 2^k points of the form $\{a_{1,1}, a_{2,1}\} \times \{a_{1,2}, a_{2,2}\} \times \cdots \times \{a_{1,k}, a_{2,k}\}$, where each $a_{i,j} \in \{0,1\}^n$.

$$\mathcal{E}(f) = E_D \left[\prod_{\overline{a}\in D} f(\overline{a}) \right].$$

Notice that the definition of $\mathcal{E}()$ for the 2-party case is recovered when $k = 2$. The next lemma is also an easy generalization.

Lemma 12.21

$$\mathit{Disc}(f) \le (\mathcal{E}(f))^{1/2^k}.$$

Proof: The proof is analogous to Lemma 12.16 and left as an exercise. The only difference is that instead of defining 2 random functions we need to define k random functions $g_1, g_2, g_k : \{0,1\}^{nk} \to \{-1, 1\}$, where g_i depends on every one of the k coordinates except the ith. ■

Now we can prove a lowerbound for the Generalized Inner Product function. Note that since we changed the range to $\{-1, 1\}$ it is now defined as

$$GIP_{k,n}(x_1, x_2, \ldots, x_k) = (-1)^{\sum_{i\le n}\prod_{j\le k} x_{ij} (mod 2)}. \tag{10}$$

Theorem 12.22
The function $GIP_{k,n}$ has k-party communication complexity $\Omega(n/8^k)$ as n grows larger.

Proof: We use induction on k. For $k \ge 1$ let β_k be defined using $\beta_1 = 0$ and $\beta_{k+1} = \frac{1+\beta_k}{2}$. We claim that

$$\mathcal{E}(GIP_{k,n}) \le \beta_k^n.$$

Assuming truth for $k-1$ we prove for k. A random cube D in $\{0,1\}^{nk}$ is picked by picking $a_{11}, a_{21} \in \{0,1\}^n$ and then picking a random cube D' in $\{0,1\}^{(k-1)n}$.

$$\mathcal{E}(GIP_{k,n}) = \mathbf{E}_{a_{11},a_{21}}\left[\mathbf{E}_{D'}\left[\prod_{\overline{a}\in\{a_{11},a_{21}\}\times D'} GIP_{k,n}(\overline{a})\right]\right] \tag{11}$$

The proof proceeds by considering the number of coordinates where strings a_{11} and a_{21} are identical. Examining the expression for $GIP_{k,n}$ in (10) we see that these coordinates contribute nothing once we multiply all the terms in the cube, since their contributions get squared and thus become 1. The coordinates that contribute are

TO BE COMPLETED ■

12.4 Probabilistic Communication Complexity

Will define the model, give the protocol for EQ, and describe the discrepancy-based lowerbound.

12.5 Overview of other communication models

We outline some of the alternative settings in which communication complexity has been studied.

Nondeterministic protocols: These are defined by analogy to **NP**. In a nondeterministic protocol, the players are both provided an additional third input z ("nondeterministic guess"). Apart from this guess, the protocol is deterministic. The *cost* incurred on x, y is

$$\min_z \quad \{|z| + \text{number of bits exchanged by protocol when guess is } z\}.$$

The *nondeterministic communication complexity* of f is the minimum k such that there is a nondeterministic protocol whose cost for all input pairs is at most k.

In general, one can consider communication protocols analogous to **NP**, **coNP**, **PH** etc.

Randomized protocols: These are defined by analogy to **RP**, **BPP**. The players are provided with an additional input r that is chosen uniformly at random from m-bit strings for some m. Randomization can significantly reduce the need for communication. For instance we can use fingerprinting with random primes (explored in Chapter 7), to compute the equality function by exchanging $O(\log n)$ bits: the players just pick a random prime p of $O(\log n)$ bits and exchange $x \pmod p$ and $y \pmod p$.

Average case protocols: Just as we can study average-case complexity in the Turing machine model, we can study communication complexity when the inputs are chosen from a distribution $\mathcal{D}$. This is defined as

$$C_{\mathcal{D}}(f) = \min_{\text{protocols } \mathcal{P}} \sum_{x,y} \Pr[(x,y)\in\mathcal{D}] \times \{\text{Number of bits exchanged by } \mathcal{P} \text{ on } x,y.\}$$

Computing a non boolean function: Here the function's output is not just $\{0, 1\}$ but an m-bit number for some m. We discuss one example in the exercises.

Asymmetric communication: The "cost" of communication is asymmetric: there is some B such that it costs the first player B times as much to transmit a bit than it does the second player. The goal is to minimize the total cost.

Multiparty settings: The most obvious generalization to multiparty settings is whereby f has k arguments $x_1, x_2, \ldots, x_k$ and player i gets x_i. At the end all players must know $f(x_1, x_2, \ldots, x_k)$. This is not as interesting as the so-called "number of the forehead" where player i can see all of the input except for x_i. We discuss it in Section **??** together with some applications.

Computing a relation: There is a relation $R \subseteq \{0, 1\}^n \times \{0, 1\}^n \times \{1, 2, \ldots, m\}$ and given $x, y \in B^n$ the players seek to agree on any $b \in \{1, 2, \ldots, m\}$ such that $(x, y, b) \in R$. See section **??**.

These and many other settings are discussed in [**?**].

12.6 Applications of communication complexity

We briefly discussed parallel computation in Chapter 6. Yao [**?**] invented communication complexity as a way to lowerbound the running time of parallel computers for certain tasks. The idea is that the input is distributed among many processors, and if we partition these processors into two halves, we may lowerbound the computation time by considering the amount of communication that must necessarily happen between the two halves. A similar idea is used to prove time/space lowerbounds for VLSI circuits. For instance, in a VLSI chip that is an $m \times m$ grid, if the communication complexity for a function is greater than c, then the time required to compute it is at least c/m.

Communication complexity is also useful in time-space lowerbounds for Turing machines (see Problem 1 in exercises), and circuit lowerbounds (see Chapter 13).

Data structures such as heaps, sorted arrays, lists etc. are basic objects in algorithm design. Often, algorithm designers wish to determine if the data structure they have designed is the best possible. Communication complexity lowerbounds can be used to establish such results. See [**?**].

Yannakakis [**?**] has shown how to use communication complexity lowerbounds to prove lowerbounds on the size of polytopes representing **NP**-complete problems. Solving the open problem mentioned in Problem 8 in the exercises would prove a lowerbound for the polytope representing vertex cover.

Exercises

§1 If $S(n) \leq n$, show that a space $S(n)$ TM takes at least $\Omega(n/S(n))$ steps to decide the language $\{x\#x : x \in \{0, 1\}^*\}$.

§2 Show that the high school definition of rank (the size of the largest set of independent rows or columns) is equivalent to that in Definition 12.9.

§3 Give a fooling set argument that proves that $C(f) \geq \lceil \log \operatorname{rank}(M(f)) \rceil$.

§4 Show that $C(f)\text{rank}(M(f) + 1$.

§5 Consider x, y as vectors over $GF(2)^n$ and let $f(x, y)$ be their inner product mod 2. Prove that the communication complexity is n.

Hint: Lowerbound the rank of the matrix $2M(f) - J$ where J is the all-1 matrix.

What field should you use to compute the rank? Does it matter?

§6 Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be such that all rows of $M(f)$ are distinct. Show that $C(f) \geq \log n$.

Hint: Lowerbound the rank.

§7 (Aho, Ullman, Yannakakis) Show that $C(f) = O(\log^2 \chi(f))$.

Hint: The players try to determine which of the $|\chi(f)|$ rectangles their input-pair lies in. The protocol has $O(\log \chi(f))$ phases, and in each phase $O(\log \chi(f))$ bits get communicated.

§8 For any graph G with n vertices, consider the following communication problem: Player 1 receives a clique C in G, and Player 2 receives an independent set I. They have to communicate in order to determine $|C \cap I|$. (Note that this number is either 0 or 1.) Prove an $O(\log^2 n)$ upperbound on the communication complexity.

Can you improve your upperbound or prove a lower bound better than $\Omega(\log n)$? (Open question)

§9 Prove Lemma 12.21 using the hint given there.

§10 (Karchmer-Wigderson) Consider the following problem about computing a relation. Associate the following communication problem with any function $f : \{0,1\}^n \to \{0,1\}$. Player 1 gets any input x such that $f(x) = 0$ and player 2 gets any input y such that $f(y) = 1$. They have to communicate in order to determine a bit position i such that $x_i \neq y_i$.

Show that the communication complexity of this problem is *exactly* the minixmum depth of any circuit that computes f. (The maximum fanin of each gate is 2.)

§11 Use the previous question to show that computing the parity of n bits requires depth at least $2 \log n$.

§12 Show that the following computational problem is in **EXP**: given the matrix $M(f)$ of a boolean function, and a number K, decide if $C(f) \leq K$.

(Open since Yao [**?**]) Can you show this problem is complete for some complexity class?

Chapter notes and history

Communication complexity was first defined by Yao [?]. Other early papers that founded the field were Papadimitriou and Sipser [?], Mehlhorn and Schmidt [?] (who introduced the rank lowerbound) and Aho, Ullman and Yannakakis [?].

The original log rank conjecture was that $C(f) = O(\mathrm{rank}(M(f)))$ but this was disproved by Raz and Spieker [?].

The book by Nisan and Kushilevitz [?] is highly recommended.