CS 2429 - Foundations of Communication Complexity

Lecturer: Toniann Pitassi

1 Rectangles

The success in proving good lower bounds on the communication complexity comes from the *combinatorial* view of protocols. The idea is to view protocols as a way to partition the space of all possible input pairs, $X \times Y$, into special sets called combinatorial rectangles. Let P be a protocol and v be a node of the protocol tree. We denote by R_v is the set of inputs (x, y) that reach node v. Let L be the set of leaves of the protocol P. It is easy to see that the set $\{R_l\}_{l \in L}$ is a *partition* of $X \times Y$. This discussion leads to the following fundamental element in the combinatorics of protocols.

Definition 1 (Rectangle). *A rectangle in $X \times Y$ is a subset $R \subseteq X \times Y$ such that $R = A \times B$ for some $A \subseteq B$ and $B \subseteq Y$.*

The connection between rectangles and protocols is implicit in the following proposition.

Proposition 1. *For all $l \in L$, the set R_l is a rectangle.*

Proof. By induction on the depth of the protocol tree. □

Moreover, by the definition of the protocol in the above rectangles the function f has a fixed value, i.e., monochromatic.

Definition 2 (f-monochromatic). *A subset $R \subseteq X \times Y$ is f-monochromatic if f is fixed*[1] *on R.*

The following two statement are immediate from the above definitions.

Fact 2. *Any protocol P for f induces a partition of $X \times Y$ into f-monochromatic rectangles. The number of (f-monochromatic) rectangles equals the number of leaves of P.*

Fact 3. *If any partition of $X \times Y$ into f-monochromatic rectangles requires at least t rectangles, then $D(f) \geq \log_2 t$.*

1.1 The Fooling Set Argument

Consider the following $2^n \times 2^n$ matrix associated with equality function $EQ(x, y)$, $|x| = |y| = n$.

$$M_{EQ} := \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}$$

[1]There exists $z \in \{0, 1\}$ such that for all $(x, y) \in R$, $f(x, y) = z$.

Each "1" has to be in its own 1-monochromatic rectangle. Thus the number of monochromatic rectangles is greater than 2^n. This observation motivates the following definition of a "fooling set".

Definition 3. *Let $f : X \times Y \to \{0,1\}$. A subset $S \subseteq X \times Y$ is a fooling set for f if there exists $z \in \{0,1\}$ such that*

(i) $\forall\ (x,y) \in S,\ f(x,y) = z$;

(ii) for any two distinct $(x_1,y_1),\ (x_2,y_2) \in S$, either $f(x_1,y_2) \neq z$ or $f(x_2,y_1) \neq z$.

Lemma 4. *If f has a fooling set S of size t, then $D(f) \geq \log_2 t$.*

1.2 The Rank Lower Bound Method

Given any boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ we can associate a $2^n \times 2^n$ matrix M_f, where $M_f(x,y) = f(x,y)$. In words, M_f specifies the values of the function f on any input $(x,y) \in X \times Y$. The rank lower bound method is an algebraic method to give lower bounds on $D(f)$ by computing the rank of M_f.

Definition 4. *For any function f, rank(f) is the linear rank of M_f over $\mathbb{R}$.*

The following lemma gives a lower bound on the deterministic communication complexity of f through the rank of M_f.

Lemma 5. *Let a function f . Then $D(f) \geq \log_2 \mathit{rank}(f)$.*

Proof. Let L_1 be the set of leaves of any protocol tree that gives output 1. For each $l \in L_1$, let M_l be a $2^n \times 2^n$ matrix which is 1 on all $(x,y) \in R_l$ and 0 otherwise. It is clear that

$$M_f = \sum_{l \in L_1} M_l.$$

Fact : The rank function is a sub-additive function, i.e., $\mathrm{rank}(A+B) \leq \mathrm{rank}(A) + \mathrm{rank}(B)$ for any matrix A, B. Therefore,

$$\mathrm{rank}(M_f) \leq \sum_{l \in L_1} \mathrm{rank}(M_l).$$

Notice that $\mathrm{rank}(M_l) = 1$ for any $l \in L_1$ since M_l can be expressed as an outer-product of two vectors[2]. Therefore $\mathrm{rank}(M_f) \leq |L_1| \leq |L|$, which implies that

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

□

[2]These vectors are the characteristic vectors for the rectangle that reaches l.

1.2.1 The Log Rank Conjecture

The above fact shows that communication complexity lower bounds can be proven from rank lower bounds. It is a longstanding open question whether or not there is a converse relationship. The best upper bound known is $D(f) \leq rank(f) + 1$, and thus the gap between these bounds is enormous. The log rank conjecture asserts that $D(f) = (\log rank(f))^{O(1)}$ for every Boolean function f. That is, it states that the deterministic communication complexity of any two party function is equal to the log of the rank of its associated matrix, up to polynomial factors. (An early paper by Lovasz and Saks is attributed to the conjecture.)

Despite much research, very little is known about this conjecture. Until recently the best upper bound known was:

$$D(f) \leq log(4/3)rank(f)$$

.

A recent paper by Ben-Sasson, Lovett and Ron-Zewi established a connection between the log-rank conjecture and a number theoretic conjecture known as the Freiman-Ruzsa conjecture:

Theorem 6. *Assuming the Freiman-Ruzsa conjecture over F_2^n, for any boolean f*

$$D(f) \leq O(rank(f)/logrank(f)).$$

Very recently, Lovett proved the following unconditional result.

Theorem 7.

$$D(f) \leq O(\sqrt{rank(f)}logrank(f)).$$

His proof is based on discrepancy of low rank matrices.

But recently, there has been renewed interest in the conjecture.

2 Covers

Definition 5. *Let $f : X \times Y \to \{0, 1\}$ be a function:*

1. $C^P(f) =$ *minimum number of leaves in a* protocol tree *for f. This is also called the protocol number.*

2. $C^D(f) =$ *minimum number of monochromatic rectangles in a (rectangular disjoint)* partition *of $X \times Y$. This is called the partition number.*

3. $C(f) =$ *minimum number of monochromatic rectangles that* covers *$X \times Y$. This called the cover number.*

4. $C^z(f) =$ *minimum number of monochromatic rectangles needed to cover the z-inputs*[3] *of f.*

The measure $C^z(f)$ has a natural interpretation in terms of the nondeterministic communication complexity of f. More precisely, the deterministic communication complexity of f, denoted by $N^1(f)$ is equal to $\log_2 C^1(f)$. And similarly, the conondeterministic communication complexity of f, $N^0(f)$ is equal to $\log_2 C^0(f)$.

[3] The z-inputs of a function f is the set $\{(x, y) \mid f(x, y) = z\}$.

Proposition 8. *For all* $f : X \times Y \to \{0, 1\}$*:*

- $C(f) \leq C^D(f) \leq C^P(f) \leq 2^{D(f)}$.
- $C(f) = C^0(f) + C^1(f)$.

Lemma 9 (Balancing Protocols)**.** *Let f a function. Then*

$$\log C^P(f) \leq D(f) \leq 2 \log_{3/2} C^P(f)$$

Proof. The lower bound on $D(f)$ is immediate. For the upper bound, it suffices to show that given any deterministic protocol P that computes f with s leaves, we are able to create a new protocol P' for f with deterministic communication complexity $O(\log s)$.

By hypothesis, we know that the protocol tree T has s leaves. The proof relies heavily on the following claim.

Claim 10. *For any tree T with s leaves, $|T| = s$, there exists a node v of T, such that for the sub-tree T_v rooted at v the following inequality holds,*

$$\frac{s}{3} \leq |T_v| \leq \frac{2s}{3}.$$

The input of the new protocol P' is the protocol tree T, the input pair (x, y), and the number of leaves s.

1. Alice and Bob determine a node v such that $\frac{s}{2} \leq |T_v| \leq \frac{2s}{3}$.
2. Both decide if $(x, y) \in R_v$[4], by sending one bit each of them, in total 2 bits.
3. If yes, recurse on the rectangle R_v.
4. If no, recurse on the tree T_{new}, where T_{new} is the same tree as T, except that the sub-tree T_v is replaced by a single node/leave with value 0.

Let's analyse the above protocol. Let $Q(s)$ be the number of bits that are communicated by the above protocol when the input tree has s leaves. It is easy to see that the following recursion on $Q(s)$ holds,

$$Q(s) \quad \leq \quad 2 + Q(\frac{2s}{3}),$$

where 2 is the bits that are communicated at the current step and $Q(2s/3)$ the number of bits that will be communicated in the next (recursive) step in worst case. Also note that $Q(1) = 0$. Applying the above inequality repeatedly we get

$$\begin{aligned} Q(s) \quad &\leq \quad \underbrace{2 + 2 + \cdots + 2}_{i} + Q(\frac{2^i s}{3^i}), \quad \text{by setting } i = \log_{3/2} s, \\ &= \quad 2 \log_{3/2} s. \end{aligned}$$

Setting $s = C^P(f)$, i.e., the minimum number of leaves for a protocol that computes f, and notice that $D(f) = Q(C^P(f))$, gives the lemma. □

[4] R_v is the rectangle that corresponds to the sub-tree T_v

The following theorem states that the deterministic communication complexity of any Boolean function is at most the product of the nondeterministic and conondeterministic communication complexities of the function.

Theorem 11. *For every function $f : X \times Y \to \{0,1\}$, $D(f) = O(N^0(f)N^1(f))$.*

Proof. There are two proofs of this theorem presented in Kushilevitz-Nisan. We present the first "algorithmic" one.

First observe that if $R = S \times T$ is a 0-monochromatic rectangle of f, and $R' = S' \times T'$ is a 1-monochromatic rectangle of f, then either $S \cap S' = \emptyset$ or $R \cap R' = \emptyset$.

Now suppose that for some input (x, y), $f(x, y) = 1$ so that (x, y) is in a 1-rectangle, R. Then either R intersects in rows with at most half of the 0-rectangles, or R intersects in columns with at most half of the 0-rectangles. (If not, then by the pigeonhole principle, there must exist some 0-rectangle that intersects with R, which is a contradiction.)

Phase 1 of the protocol is as follows. Alice and Bob will try to find a 1-rectangle that could be the rectangle containing R' and that intersects with at most half of the 0-rectangles in rows, or that intersects with at most half of the 0-rectangles in columns. If (x, y) is in a 1-rectangle, then such an R' will always exist (for R' equal to the rectangle containing (x, y) it will have this property) and so they wll find it. Thus, if no such rectangle is found, then they can conclude that (x, y) cannot be in a 1-rectangle, so they can conclude that $f(x, y) = 0$. To carry this out, first Alice looks for a 1-rectangle that contains row x and that intersects in rows with at most half of the 0-rectangles. If she finds such a rectangle, R', she sends its name to Bob and this phase then terminates. If she fails to find such a 1-rectangle, then Bob looks for a 1-rectangle that contains row y and that intersects in columns with at most half of the 0-rectangles. If he finds such a rectangle R', he sends its name to Alice, and then this phase terminates. If neither find such a rectangle, then they conclude $f(x, y) = 0$ and the protocol ends. If the protocol does not end after phase 1, then they have a rectangle R' that either intersects in columns with at most half of the 0-rectangles or intersects in rows with at most half of the 0-rectangles. This allows them to prune the space of possible 0-rectangles: If R' intersects in rows with at most half of the 0-rectangles, remove those 0-rectangles that do not intersect in rows; otherwise if R' intersects in columns with at most half of the 0-rectangles, then remove those 0-rectangles that do not intersect in columns. This removes at least half of the 0-rectangles altogether.

In phase i, they continue as above. Alice considers the set of all 0-rectangles that are still under consideration at phase i. If there are no such rectangles she outputs $f(x, y) = 1$. Otherwise, she looks for a 1-rectangle that contains row x and that intersects in rows with at most half of the 0-rectangles that are still under consideration. If she finds such a rectangle, R', she sends its name to Bob and the phase is completed. The set of 0-rectangles that remain active for the next phase $i+1$ are those that intersect in rows with R'. Otherwise she tells Bob that no such rectangle exists.

If phase i hasn't terminated yet, then Bob looks for a 1-rectangle that contains column y and that intersections in columns with at most half of the 0-rectangles that are still under consideration. If such a rectangle, R' exists, then Bob sends its name to Alice and the phase is completed. The set of 0-rectangles that remain active for the next phase $i + 1$ are those that intersect in columns with Q. Otherwise Bob outputs $f(x, y) = 0$.

For runtime, the number of phases is at most log of the number of 0-rectangles ($\log N^0(f)$), and at each phase they send $O(\log N^1(f))$ many bits, for a total of $O(\log N^0(f) \log N^1(f))$ bits.

For correctness, first suppose that (x, y) is in a 1-rectangle. Then at each phase, Alice and Bob will find R' and thus the protocol will run for all phases until all 0-rectangles have been eliminated, at which point the protocol will say $f(x, y) = 1$. Next suppose that (x, y) is in a 0-rectangle, Q. Then we claim that this 0-rectangle Q will never be pruned/removed, and thus at some phase they must not find a good R' and therefore they will output $f(x, y) = 0$ at this phase. Suppose that (x, y) is in the 0-rectangle, Q. There are two ways for Q to be removed: (i) there is some 1-rectangle R' that contains x and that does not intersect Q in any rows – but this is impossible since it must intersect Q in row x; (ii) there is a 1-rectangle R' that contains y and that does not intersect Q in any columns – again impossible since they intersect in row y. $\square$