Communication Complexity: Lecture 2

Nathan Harms

January 16, 2026

1 Recap & Warm-Up

To help recall the previous lecture, let's define a new communication problem.

Definition 1.1 (Greater-Than)**.** The Greater-Than communication problem on integers $i, j \in [N]$ is defined as

$$\mathsf{GT}_N(i,j) = \begin{cases} 1 & \text{if } i > j \\ 0 & \text{if } i \leq j \end{cases}$$

Using what we learned in the previous lecture, we can answer the following question:

Exercise 1.2. *What is the communication complexity of* Greater-Than*, and what is its matrix form?*

2 Rank

The topic of this lecture is the relationship between communication complexity and *rank*. Recall several definitions of the rank of a matrix.

Theorem 2.1. *Let $M \in \mathbb{R}^{m \times n}$ be any matrix and define the following.*

1. *Let r_1 be the maximum size $|R|$ of a set $R \subseteq [m]$ such that rows R of M are linearly independent.*
2. *Let r_2 be the maximum size $|C|$ of a set $C \subseteq [n]$ such that columns C of M are linearly independent.*
3. *Let r_3 be the minimum value such that there exist $U \in \mathbb{R}^{m \times r}$ and $U \in \mathbb{R}^{r \times n}$ such that $M = UV$.*
4. *Let r_4 be the minimum value such that there exist vectors $u_1, \dots, u_r \in \mathbb{R}^m$ and $v_1, \dots, v_r \in \mathbb{R}^n$ that satisfy $M = \sum_{i=1}^r u_i v_i^\top$.*

Then $r_1 = r_2 = r_3 = r_4$.

This quantity is the rank of M, denoted $\text{rank}(M)$.

Proof. Exercise for the reader. ■

From this proof we can deduce some extra facts about boolean matrices $M \in \{0,1\}^{m\times n}$ that are useful for communicaton complexity.

Corollary 2.2. *Let $M \in \{0,1\}^{m\times n}$. Then:*

1. $\mathrm{rank}(M)$ *is the minimum r such that there exists a submatrix $V \in \{0,1\}^{r\times n}$ of M, and $U \in \mathbb{R}^{m\times r}$, such that $M = UV$.*
2. $\mathrm{rank}(M)$ *is the minimum r such that there exists a submatrix $U \in \{0,1\}^{m\times r}$ of M, and $V \in \mathbb{R}^{r\times n}$, such that $M = UV$.*

Theorem 2.3. *There is an algorithm which computes the rank of a given matrix $M \in \mathbb{R}^{n\times n}$ using at most $O(n^3)$ arithmetic operations.*

3 Lower Bounds via Rank

Let's see how to prove lower bounds on communication complexity using the rank of a matrix.

Theorem 3.1. *For any $M \in \{0,1\}^{m\times n}$ that is not the all-1s matrix,*

$$\log(\mathrm{rank}(M) + 1) \le \mathsf{D}(M).$$

Before proving the theorem, recall the standard transformation between sets and boolean vectors:

Definition 3.2. Let $S \subseteq [n]$ be any set. We say a vector $u \in \{0,1\}^n$ *indicates* S if

$$\forall i \in [n]: \qquad x_i = 1 \iff i \in S.$$

Proof of Theorem 3.1. We will show that $\mathrm{rank}(M) + 1 \le 2^{\mathsf{D}(M)}$. Recall that $\chi_1(M) \le 2^{\mathsf{D}(M)}$, so for $k = \chi_1(M)$ there exists a collection $\mathcal{P}_1 = \{R_1, \ldots, R_k\}$ of disjoint 1-monochromatic rectangles $R_\ell = X_\ell \times Y_\ell$ which partition the 1-entries of M.

For each $\ell \in [k]$, let $u_\ell \in \{0,1\}^m, v_\ell \in \{0,1\}^n$ be the vectors which indicate the sets X_ℓ and Y_ℓ respectively. By definition, the matrix $u_\ell v_\ell^\top$ satisfies

$$(u_\ell v_\ell^\top)_{i,j} = \mathbb{1}\left[i \in X_\ell\right] \cdot \mathbb{1}\left[j \in Y_\ell\right] = \mathbb{1}\left[(i,j) \in R_\ell\right].$$

Since the rectangles R_ℓ partition the 1-entries of M, we have

$$M = \sum_{\ell=1}^{k} u_\ell v_\ell^\top.$$

Therefore, by the 4th definition of rank, $\mathrm{rank}(M) \le k = \chi_1(M)$. Since M is not the all-1s matrix, $\chi(M) \ge \chi_1(M) + 1$, so

$$\log(\mathrm{rank}(M) + 1) \le \log(\chi(M)) \le \mathsf{D}(M).$$ ■

We can use Theorem 3.1 to prove lower bounds on communication complexity. To give an example, let's define one of the most important communication problems.

Definition 3.3 (Disjointness)**.** The function $\mathrm{Disj}_n \colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is defined as

$$\mathrm{Disj}_n(x,y) = \begin{cases} 0 & \text{if } \exists i \in [n] \colon x_i = y_i = 1 \\ 1 & \text{otherwise.} \end{cases}$$

In other words, $\mathrm{Disj}_n(x,y) = 1$ if x and y indicate disjoint sets. (We say $x \in \{0,1\}^n$ *indicates a set* $S \subseteq [n]$ when $x_i = 1$ if and only if $i \in S$.)

This problem is one of the most important for applications that we will see later. Let's prove an optimal lower bound on the communication complexity of Disjointness:

Theorem 3.4. $\mathsf{D}(\mathrm{Disj}_n) = n + 1$.

To prove this theorem, we'll use a fact about rank. First define the tensor product of two matrices $A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{m' \times n'}$, denoted $A \otimes B \in \mathbb{R}^{mm' \times nn'}$, as

$$A \otimes B := \begin{bmatrix} A_{11}B & A_{12}B & \cdots & A_{1n}B \\ A_{21}B & A_{22}B & \cdots & A_{2n}B \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1}B & A_{m2}B & \cdots & A_{mn}B \end{bmatrix}$$

Rank behaves well with respect to the tensor product:

Fact 3.5. $\mathrm{rank}(A \otimes B) = \mathrm{rank}(A) \cdot \mathrm{rank}(B)$.

Proof of Theorem 3.4. An upper bound holds from the general communication upper bound of $n+1$. For the lower bound, we will show that the matrix has full rank. Write $D_n \in \{0,1\}^{2^n \times 2^n}$ for the communication matrix of Disj_n. Observe that

$$D_1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

and therefore $\mathrm{rank}(D_1) = 2$. Now consider D_n for $n > 1$. We may assume that the rows and columns are both ordered lexicographically, i.e. $0000, 0001, 0010, \ldots, 1111$. In that case, the communication matrix may be written

$$D_n = \begin{bmatrix} D_{n-1} & D_{n-1} \\ D_{n-1} & 0 \end{bmatrix} = D_1 \otimes D_{n-1}.$$

Therefore, by Fact 3.5,

$$\mathrm{rank}(D_n) = 2 \cdot \mathrm{rank}(D_{n-1}).$$

By induction, we conclude that $\mathrm{rank}(D_n) = 2^n$. So

$$\mathsf{D}(\mathrm{Disj}_n) \geq \log(\mathrm{rank}(D_n) + 1) > n. \qquad \blacksquare$$

4 Is the rank lower bound optimal?

Let's say you want to prove a lower bound for your favorite problem. Now you know you can try to calculate the rank. But you may wonder, is it possible that this technique will fail?

The rank technique would fail if both of the following were true:

1. rank(M) is small; but
2. $\mathsf{D}(M)$ is big.

Can this happen?

One of the most famous open problems in communication complexity is to answer this question. There is a conjecture that the answer is (essentially) *no, this can't happen*:

Conjecture 4.1 (Log-Rank Conjecture)**.** *There exists a constant c such that, for every matrix $M \in \{0,1\}^{m\times n}$,*

$$\mathsf{D}(M) \le O(\log^c(\operatorname{rank} M)).$$

This conjecture is from László Lovász and Mike Saks in 1988 [LS88]. If the conjecture is true, meaning that rank would always tell you the communication complexity, up to a polynomial factor. Recall from last time the open problem of *computing* an approximation $\mathsf{D}(M)$ in polynomial-time. If the log-rank conjecture is true, we can accomplish this by computing the rank.

5 Progress towards the log-rank conjecture

It may not even be obvious that there is any function ϕ for which $\mathsf{D}(M) \le \phi(\operatorname{rank}(M))$. Let's start with this, and then catch up to the state of the art on this conjecture.

5.1 $\mathsf{D}(M) \le \operatorname{rank}(M)$

Theorem 5.1. *For all $M \in \{0,1\}^{m\times n}$, $\mathsf{D}(M) \le \operatorname{rank}(M) + 1$.*

The main step in this proof is the following, which will also be useful later.

Proposition 5.2. *Let $M \in \{0,1\}^{m\times n}$. Then the number of distinct columns in M is at most $2^{\operatorname{rank}(M)}$. The same bound holds for the number of distinct rows.*

Proof. Let $r = \operatorname{rank}(M)$. Recall from Corollary 2.2 that we may write

$$M = UV$$

where $U \in \mathbb{R}^{m\times r}$ and $V \in \{0,1\}^{r\times n}$. Observe that V contains at most 2^r distinct columns, because it is a boolean matrix with r rows.

Column i of M is $UV_{(\cdot,i)}$ where $V_{(\cdot,i)}$ denotes the i^{th} column of V. Therefore, if columns i, j of M are distinct, columns i, j of V must also be distinct. We conclude that there are at most 2^r distinct columns in M, since there are at most 2^r distinct columns in V. ■

We may now prove $\mathsf{D}(M) \le \operatorname{rank}(M) + 1$.

Proof of Theorem 5.1. Since $M \in \{0,1\}^{m\times n}$ has at most 2^r distinct rows, for $r = \operatorname{rank}(A)$, we may assign each row a number in $\{0,\dots,2^r\}$ indicating its equivalence class.

On input $x, y \in \{0,1\}^n$, Alice uses r bits to sends the number in $\{0,\dots,2^r\}$ that indicates its equivalence class. Bob may now determine the entry in the matrix and send it to Alice. ■

5.2 Some sad progress

We would like to improve exponentially on our bound $\mathsf{D}(M) \leq \text{rank}(M) + 1$ in order to prove the log-rank conjecture. We don't know how to do that. In fact we are not even close. The best upper bound that we know is from Benny Sudakov and Istvan Tomon in 2024:

Theorem 5.3 ([ST25]). *For all boolean matrices M, $\mathsf{D}(M) = O(\sqrt{\text{rank}(M)})$.*

This follows a breakthrough of Shachar Lovett in 2014:

Theorem 5.4 ([Lov14]). *For all boolean matrices M, $\mathsf{D}(M) = O(\sqrt{\text{rank}(M)}\log(\text{rank}(M)))$.*

We will look at an easier proof giving $\mathsf{D}(M) = O(\sqrt{\text{rank}(M)}\log^2(\text{rank}(M)))$ below.

What about trying to disprove the conjecture? Mika Göös, Toni Pitassi, and Thomas Watson showed in 2018:

Theorem 5.5 ([GPW18]). *There exist boolean matrices M such that $\mathsf{D}(M) = \widetilde{\Omega}(\log^2(\text{rank}(M)))$.*

A lot of work yet remains...

5.3 An equivalent conjecture

All of the upper bounds mentioned above have used an equivalent version of the conjecture due to Noam Nisan and Avi Wigderson in 1995 [NW95]. This equivalent conjecture says that small-rank matrices have very large monochromatic rectangles:

Conjecture 5.6. *There exists a constant c such that every $M \in \{0,1\}^{m\times n}$ with $\text{rank}(M) = r$ has a monochromatic rectangle with $mn \cdot 2^{-O(\log^c(r))}$ entries.*

Observe that any boolean matrix $M \in \{0,1\}^{m\times n}$ must have a monochromatic rectangle with at least $mn \cdot 2^{-\mathsf{D}(M)}$, because each leaf of the protocol is a monochromatic rectangle and there are at most $2^{\mathsf{D}(M)}$ leaves. This means that the log-rank conjecture implies Conjecture 5.6.

On the other hand, we will show that Conjecture 5.6 also implies the log-rank conjecture, so they are equivalent. Indeed, every improved upper bound on $\mathsf{D}(M)$ has worked by finding a large monochromatic rectangle.

Theorem 5.7 ([NW95]). *Suppose that every boolean matrix $M \in \{0,1\}^{m\times n}$ of rank r has a monochromatic rectangle containing $mn \cdot 2^{-q(r)}$, where q is any non-decreasing function. Then for all boolean matrices M of rank r,*

$$D(M) = O(q(r)\log r).$$

As a consequence, Conjecture 4.1 is equivalent to Conjecture 5.6.

Actually, they proved the bound

$$\mathsf{D}(M) = O\left(\log^2 r + \sum_{i=1}^{\log r} q\left(\frac{r}{2^i}\right)\right),$$

which is necessary to obtain the bounds in Theorems 5.3 and 5.4, but this is not necessary for our purposes. To prove our theorem, we need to recall a theorem from last time:

Theorem 5.8. *Let M be a boolean matrix and suppose there is a communication protocol computing M with ℓ leaves. Then $\mathsf{D}(M) = O(\log \ell)$.*

Proof of Theorem 5.7. Let $M \in \{0,1\}^{m\times n}$ have rank r. First, we may assume without loss of generality that $mn \le 2^{2r}$. This is because of Proposition 5.2; we may delete duplicate rows and columns without changing the rank or the communication cost.

Our goal is to design a protocol computing M with a small number of leaves, and then apply Theorem 5.8. By assumption, M contains a monochromatic rectangle with at least $mn \cdot 2^{-q(r)}$ entries. Then, by permuting M, we may write

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where A is a monochromatic rectangle with $mn \cdot 2^{-q(r)}$ entries. Then our communication protocol will be simple. On inputs $x, y \in [m] \times [n]$:

1. In the base case, fix any constant c. If $\operatorname{rank}(M) \le c$ for any constant c, or $\min\{m, n\} \le 2^c$, then we can compute M using at most $c + 1$ bits of communication.

2. If $\operatorname{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) \le \operatorname{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right)$, we partition the rows of M as

$$M = \begin{bmatrix} A & B \\ \hline C & D \end{bmatrix}$$

   and Alice sends 1 bit to Bob indicating whether her row x is in the top part or the bottom part.

3. If $\operatorname{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) > \operatorname{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right)$, we partition the columns of M as

$$M = \begin{bmatrix} A & | & B \\ C & | & D \end{bmatrix}$$

   and Bob sends 1 bit to Alice indicating whether his column y is in the left part or the right part.

4. Recurse on whichever part of M the entry (x, y) is contained in.

This protocol will terminate in some number of rounds, call it k, because the size of the matrix decreases in each iteration. It will be correct because it terminates in case (1). Now, we calculate the number of leaves in this protocol.

Consider any input (x, y). We will assign a string $t(x, y)$ that describes the operation of the protocol on (x, y). For each round $i \in [k]$, we assign:

- Symbol R if we recursed on the lower-rank part of the matrix;
- Symbol S if we recursed on the higher-rank part of the matrix;
- Symbol T at the final round k, when we reached step 1 of the protocol.

The operation of a protocol is therefore described by a string $t(x, y) \in \{R, S\}^{k-1}$ (we are ignoring the terminal T symbol which is the same for every input). To bound the number of leaves, we must bound the number of possible strings. The first step is to bound the number of R and S symbols, which we do using the following two claims:

Claim 5.9. *For all x, y, the number of S symbols in $t(x, y)$ is at most $2r \cdot 2^{q(r)} \cdot \ln(2)$.*

Proof of claim. Matrix M begins with mn entries. Let $M^{(i)}$ denote the matrix after round i, so $M^{(0)} = M$ and $M^{(k-1)}$ has either rank $\leq c$ or number of entries $\leq 2^c$, since step 1 of the protocol is executed in round k. Observe that $\text{entries}(M^{(i)}) \leq \text{entries}(M^{(i-1)})$, where $\text{entries}(A)$ denotes the total number of entries in the matrix A.

Now observe that, if round i is labelled S, it must be the case that

$$\text{entries}(M^{(i)}) \leq \text{entries}(M^{(i-1)}) \cdot (1 - 2^{-q(r)}).$$

This is because $M^{(i)}$ is obtained by removing part of the matrix containing the monochromatic rectangle A which contains at least $\text{entries}(M^{(i-1)}) \cdot 2^{-q(r)}$ entries (observe that $\text{rank}(M^{(i-1)}) \leq \text{rank}(M) = r$.

Therefore, if b is the number of S symbols in $t(x, y)$, we have

$$1 \leq \text{entries}(M^{(k-1)}) \leq \text{entries}(M) \cdot (1 - 2^{-q(r)})^b.$$

Using the inequality $1 - x \leq e^{-x}$, we have $1 \leq mn \cdot e^{-2^{-q(r)}b}$. So,

$$b \leq 2^{q(r)} \ln(mn) \leq 2^{q(r)} \ln(2^{2r}) = 2r \cdot 2^{q(r)} \cdot \ln(2),$$

where we use the bound $mn \leq 2^{2r}$. ■

Next, we need to bound the number of R symbols.

Claim 5.10. *For all x, y, the number of R symbols in $t(x, y)$ is at most $O(\log r)$.*

Proof. It will suffice to prove that, when A is a rank-1 matrix,

$$\text{rank}\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix}\right) \geq \text{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) + \text{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) - 3. \tag{1}$$

If we can prove Equation (1), then we can complete the proof as follows. Suppose the i^{th} round of $t(x, y)$ is labelled R. Then we have

$$M^{(i-1)} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where A is a monochromatic rectangle. Since A is a monochromatic rectangle, it is rank 1. Suppose $\text{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) \leq \text{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right)$, so $M^{(i)} = \begin{bmatrix} A & B \end{bmatrix}$. Then from Equation (1),

$$\text{rank}(M^{(i)}) \leq \frac{1}{2}\left(\text{rank}(M^{(i-1)}) + 3\right) \leq \frac{3}{4}\text{rank}(M^{(i-1)})$$

since $\text{rank}(M^{(i-1)}) \geq 12$ (if we choose $c \geq 12$ in step 1 of the protocol). A similar argument holds if $\text{rank}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) \leq \text{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right)$. Now, similar to the proof of the earlier claim for R symbols, if b is the number of R symbols in $t(x, y)$, then

$$1 \leq \text{rank}(M^{(k-1)}) \leq \left(\frac{3}{4}\right)^b \text{rank}(M),$$

so $b = O(\log r)$.

Now we must prove Equation (1). First, observe that

$$\text{rank}\left(\begin{bmatrix} 0 & B \\ C & D \end{bmatrix}\right) \leq \text{rank}\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix}\right) + 1,$$

which follows from the 4th definition of rank (an exercise left to the reader). Using the 1st definition of rank, one can also show

$$\text{rank}\left(\begin{bmatrix} 0 & B \\ C & D \end{bmatrix}\right) \geq \text{rank}(B) + \text{rank}(C).$$

(Again, an exercise.) Finally, using the 4th definition of rank again,

$$\text{rank}(B) = \text{rank}\left(\begin{bmatrix} 0 & B \end{bmatrix}\right) \geq \text{rank}\left(\begin{bmatrix} A & B \end{bmatrix}\right) - 1,$$

and similarly $\text{rank}(C) \geq \text{rank}\left(\begin{bmatrix} 0 \\ C \end{bmatrix}\right) - 1$. Putting these together gives Equation (1). ■

We may now bound the number of leaves of the protocol. Each leaf is assigned a string in $\{R, S\}^k$ where the number of S symbols is at most $2r \cdot 2^{q(r)} \cdot \ln(2)$ and the number of R symbols is at most $O(\log r)$. Therefore $k = 2r \cdot 2^{q(r)} \cdot \ln(2) + O(\log r)$ and the number of strings (i.e. leaves) is at most

$$\binom{k}{O(\log r)} \leq k^{O(\log r)}.$$

Then by rebalancing the protocol, we obtain

$$\mathsf{D}(M) = O\left(\log\left(k^{O(\log r)}\right)\right) = O\left(\log(r) \cdot \log\left(r \cdot 2^{q(r)}\right)\right) = O\left(q(r)\log(r)\right).$$ ■

References

[GPW18] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM Journal on Computing*, 47(6):2435–2450, 2018.

[Lov14] Shachar Lovett. Communication is bounded by root of rank. In *Proceedings of the Symposium on Theory of Computing (STOC)*, 2014.

[LS88] László Lovász and Michael Saks. Lattices, mobius functions and communications complexity. In *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 1988.

[NW95] Noam Nisan and Avi Wigderson. On rank vs. communication complexity. *Combinatorica*, 15(4):557–565, 1995.

[ST25] Benny Sudakov and István Tomon. Matrix discrepancy and the log-rank conjecture. *Mathematical programming*, 212(1):567–579, 2025.