Advanced Topics in Theoretical Computer Science

University of BRISTOL

Lecture 15

Lower Bounds 1: Communication Complexity and Streaming

Impossibility Results

How can we prove that a streaming algorithm requires at least a certain amount of space?

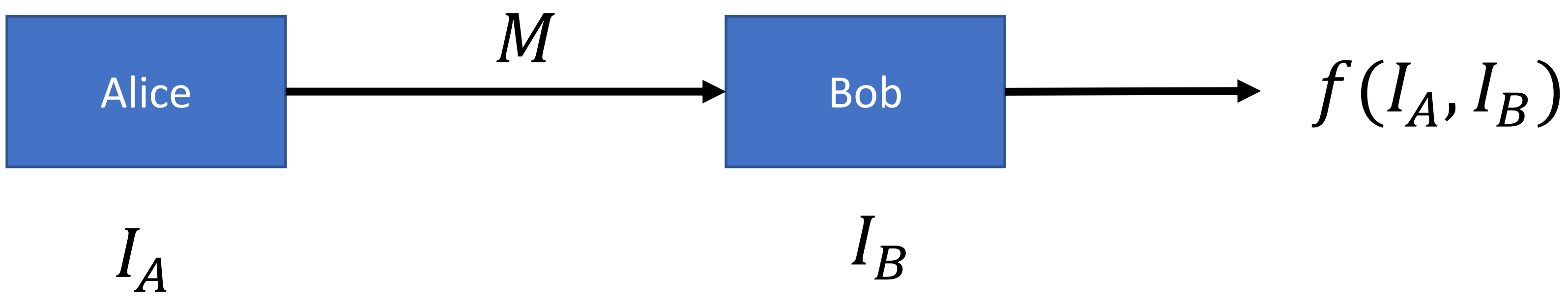
Lower Bounds = Impossibility Results:

- Computing a spanning tree requires $\Omega(\mathrm{n} \log \mathrm{n})$ space
- Computing a perfect/maximum matching requires $\Omega(n^2)$ space
- Determining the most frequent item requires $\Omega(n)$ space
- …

Communication Complexity!

Communication Complexity

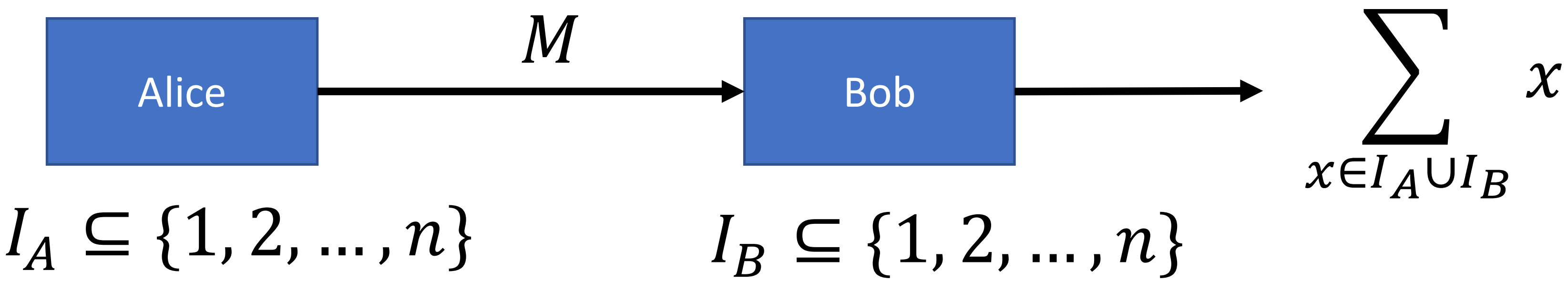
One-way Two-party Communication Model:



- Input $I_A \dot{\cup} I_B$ shared between two parties, denoted Alice and Bob
- Objective: Compute a function $f(I_A, I_B)$
- Alice sends a single message M to Bob
- Upon receipt of M, Bob outputs the result of the protocol

Goal: Ideally, $|M| \ll |I_A|$ or prove that this is not possible!

Communication Complexity: Example



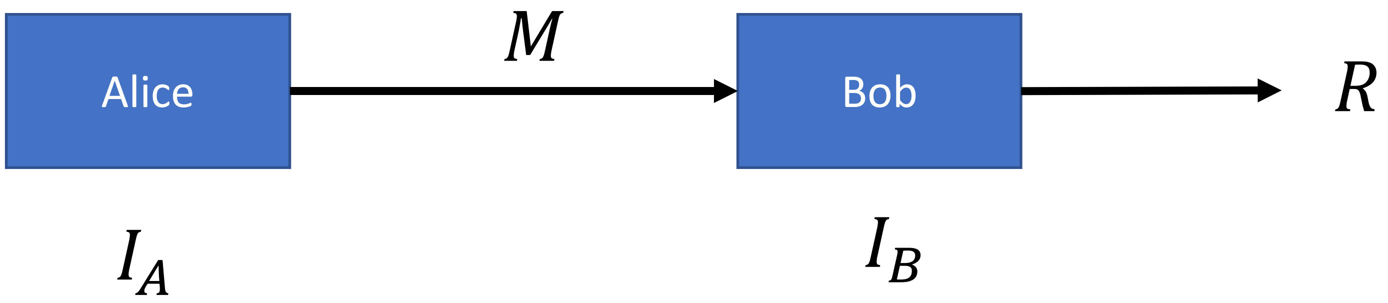
Example:

Compute the sum

Protocol:

- Alice sends the sum of her elements to Bob, Bob adds his elements
- Then: $|M| = O(\log n)$, while $|I_A|$ may be as large as n
- *Observe:* Bob does not learn much about Alice's input!

Deterministic Communication Complexity



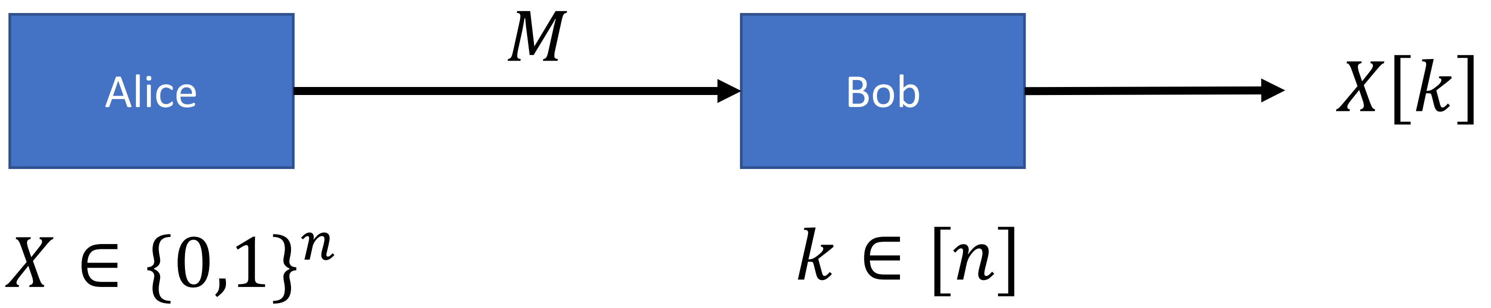
Deterministic One-way Communication Complexity:

- M is a function of I_A, i.e., $M = M(I_A)$
- The output R is a function of M and I_B, i.e., $R = R(M, I_B)$

} protocol

- Let Π be a protocol for a problem P. The *cost* of protocol Π is the maximum number of bits communicated in an execution of Π
- *The deterministic one-way communication complexity* of a problem P on instances of size n, denoted $D(P_n)$, is the minimum cost over all protocols for P

Deterministic CC of INDEX

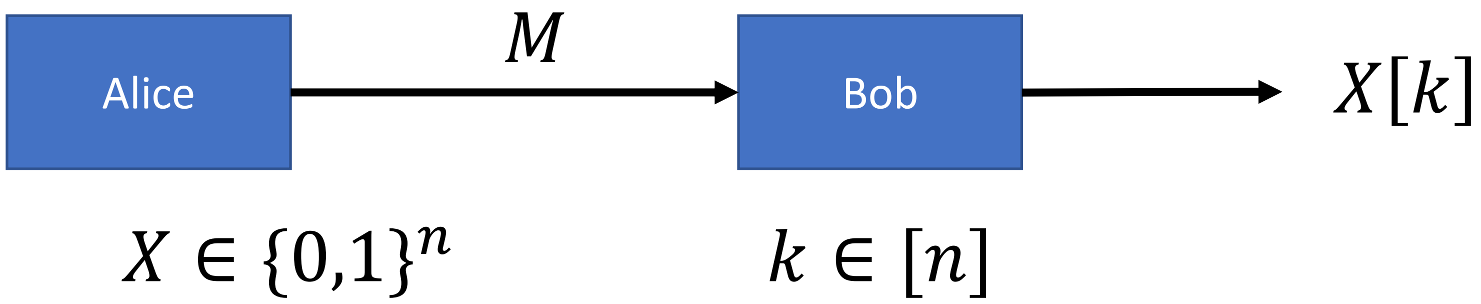


Communication Problem $\mathbf{Index}_n$:

- Alice holds $X \in \{0, 1\}^n$, Bob holds index $k \in [n]$
- Bob needs to output the bit of X at position k, i.e., $X[k]$

Goal: Determine $D(\mathrm{Index}_n)$

Deterministic CC of INDEX



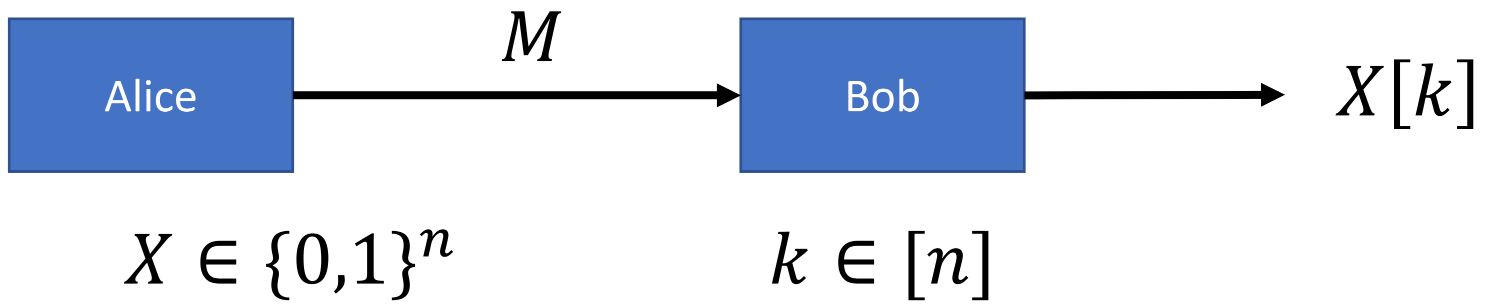
Theorem. $D(\text{Index}_n) \geq n$.

Proof.

- Let Π be an arbitrary protocol for Index_n with cost c
- Observe: Π sends at most 2^c different messages from Alice to Bob
- Observe: There are 2^n different inputs for Alice
- Suppose $c \leq n-1$. $\Rightarrow$ exist inputs $X_1, X_2 \in \{0,1\}^n$ so that both inputs yield same message m
- Since $X_1 \neq X_2$, there is a position $j \in [n]$ such that $X_1[j] \neq X_2[j]$
- Observe: output of the protocol is identical on inputs (X_1, j) and (X_2, j)
- Π therefore makes an error in one of the two cases, a contradiction to assumption $c \leq n-1$.

$\square$

Deterministic CC of INDEX



Theorem. $D(\text{Index}_n) \leq n$.

Proof.

Alice sends X to Bob, which requires a message of size n.

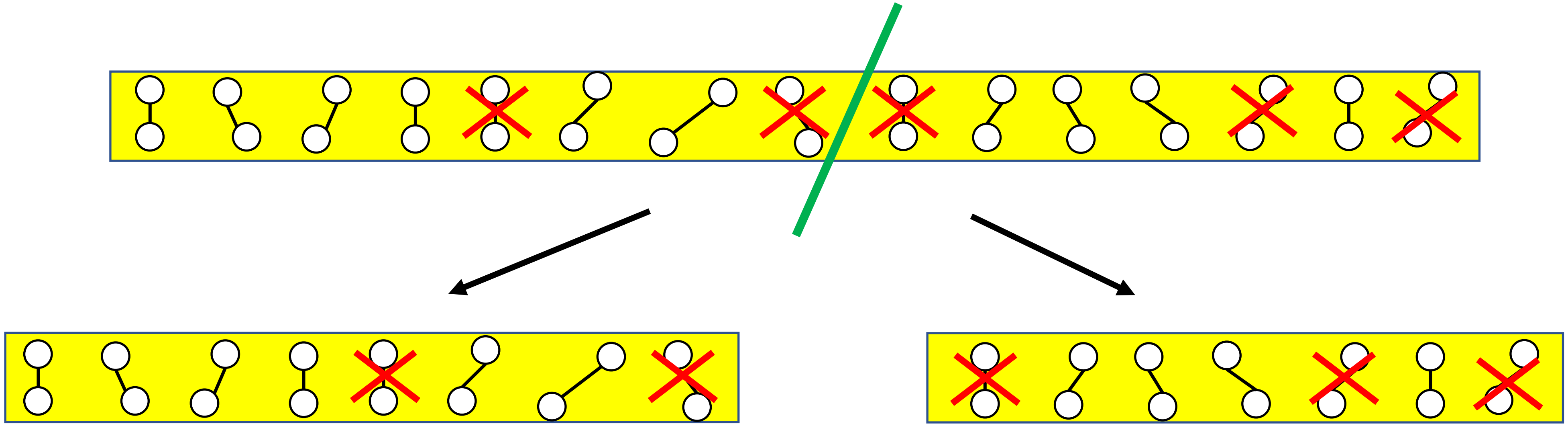
$\square$

Corollary. $D(\text{Index}_n) = n$.

One-way Communication Complexity and Streaming

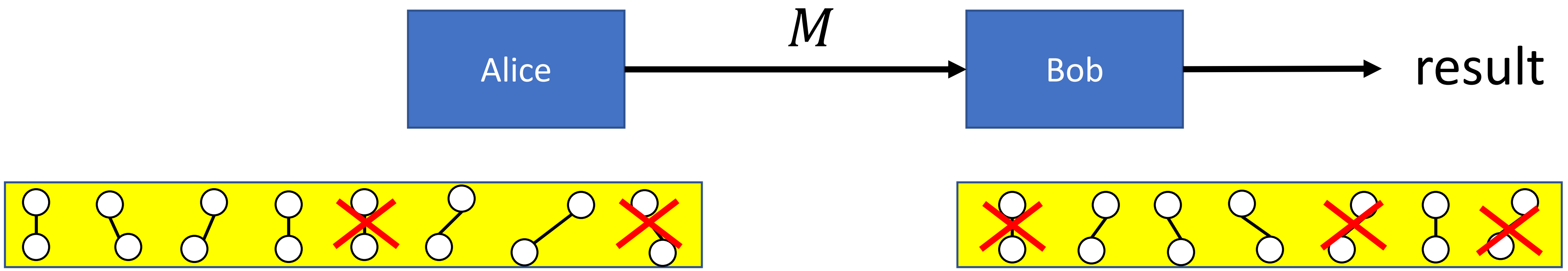
Streaming Algorithms are One-way Communication Protocols!

1. Split Input Stream into Two Parts



One-way Communication Complexity and Streaming

2. Set Two Parts as Input to Two-party Communication Problem

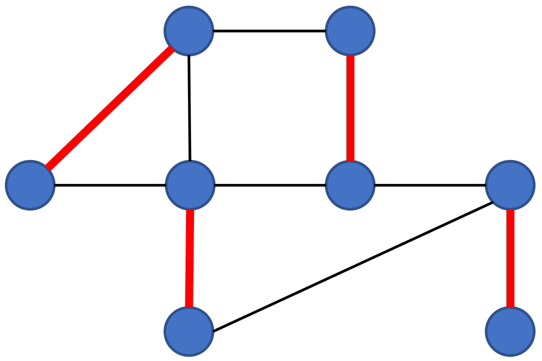


3. Reduction: Streaming Algorithm A with space s yields Communication Protocol with cost s!

- Alice runs A on her part of the input (stream)
- Message M consists of memory state of A (size at most s)
- Bob continues A on his part of the input and outputs result!

Our 1st Streaming Lower Bound: Maximum Matching

Maximum Matching:



Maximum Matching M^*

Goal: One-pass streaming algorithm for computing a Maximum Matching (no approximation!)

We will prove: Any deterministic streaming algorithm for Maximum Matching requires space $\Omega(n^2)$, where n is the number of vertices of the input graph.

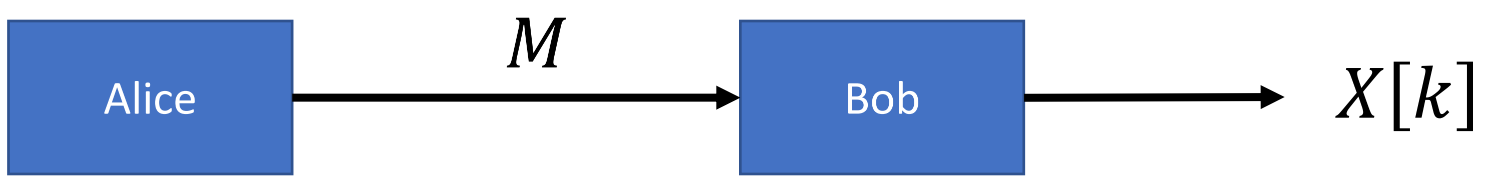
Our 1st Streaming LB: Maximum Matching

Theorem. Every deterministic streaming algorithm for Maximum Matching requires space $\Omega(n^2)$, where n is the number of vertices of the input graph.

Proof.

- Let $\boldsymbol{A}$ be a one-pass deterministic streaming algorithm for Maximum Matching with space $s(n)$ (on an n-vertex graph)
- We will show that using $\boldsymbol{A}$ we can construct a communication protocol Π for $\mathrm{Index}_{n^2/16}$ with message size $s(n)$
- Since $\mathrm{D}\left(\mathrm{Index}_{\frac{n^2}{16}}\right) \geq \frac{n^2}{16}$, we have $s(n) = \Omega(n^2)$.

Our 1st Streaming LB: Maximum Matching (2)



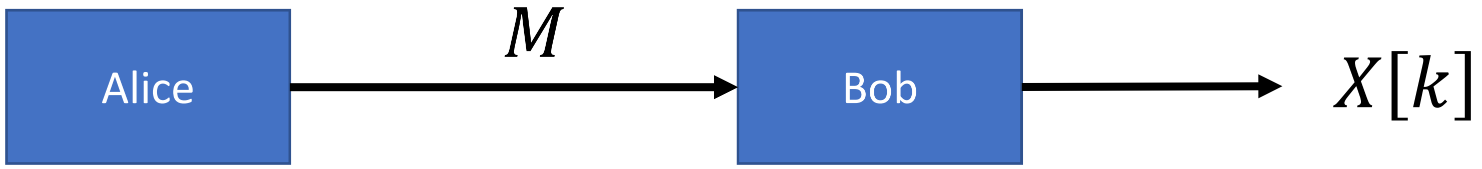
$X \in \{0,1\}^{\frac{n^2}{16}}$ $\qquad$ $k \in [\frac{n^2}{16}]$

Proof. (continued)

- Construction: Let (X, k) be an instance of $\mathrm{Index}_{\frac{n^2}{16}}$
- Alice and Bob construct a joint graph $G = G_1 \cup G_2$
- Let $f: \left[\frac{n}{4}\right] \times \left[\frac{n}{4}\right] \rightarrow [\frac{n^2}{16}]$ be an arbitrary bijection ($[x] :=$ {1,2, …, x})
- Alice constructs a bipartite graph $G_1 = (A_1, B_1, E_1)$, with $\mathrm{A}_1 = \mathrm{B}_1 = [\frac{n}{4}]$ and edge $(i, j) \in E_1 \Leftrightarrow X[f(i, j)] = 1$

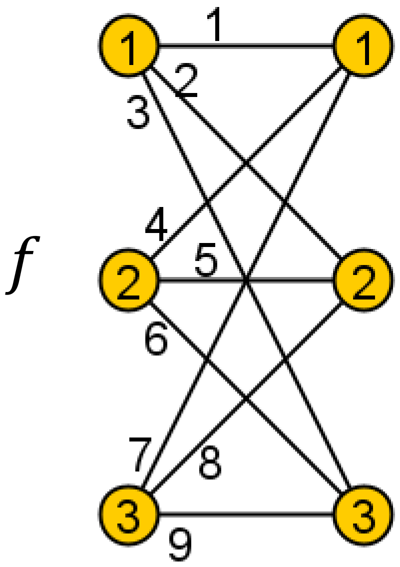
Our 1st Streaming LB: Maximum Matching (3)

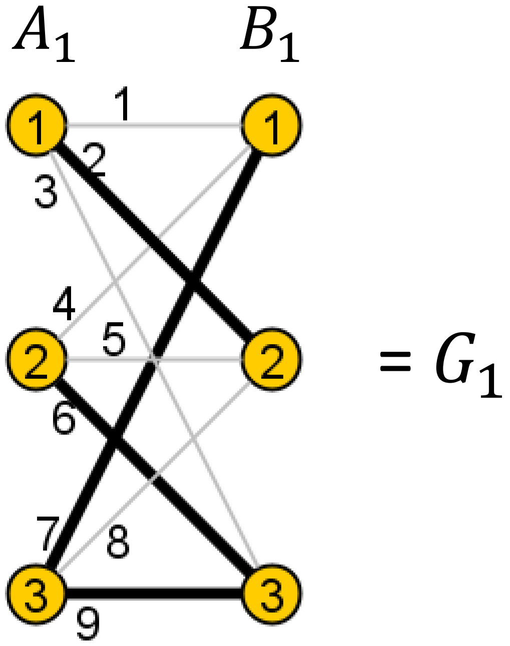
Example Construction: $(\boldsymbol{n} = \mathbf{12})$



$X = 0\,1\,0\,0\,0\,1\,1\,0\,1$
$\in \{0,1\}^{\frac{n^2}{16}} = \{0,1\}^9$

$k = 5 \in \left[\frac{n^2}{16}\right] = [9]$





Observe: $X[5] = 0$, hence edge $(2,2) \notin E_1$ $(f(2,2) = 5)$

Our 1st Streaming LB: Maximum Matching (4)

Proof. (continued)

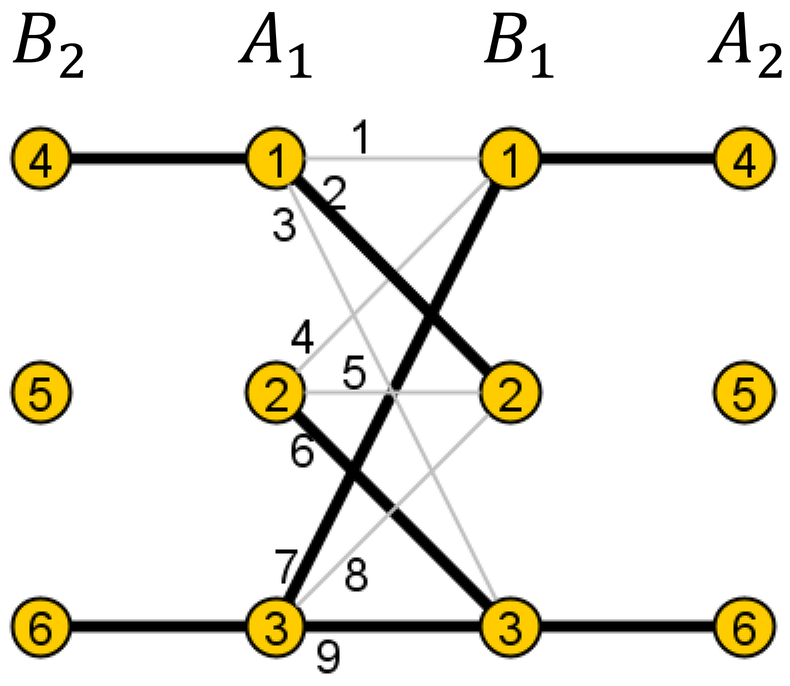
- Alice runs algorithm $\boldsymbol{A}$ on graph E_1 and sends memory state to Bob
- Bob constructs graph G_2 as follows:

1. Let $(a, b) \in A_1 \times B_1$ be such that $f(a, b) = k$
2. Define $G_2 = (A_1 \cup A_2, B_1 \cup B_2, E_2)$ with $\mathrm{A}_2 = \mathrm{B}_2 = [\frac{n}{4} + 1, \frac{n}{2}]$ and

$$E_2 = \left\{\left(\frac{n}{4} + \ell, \ell\right) \in A_2 \times B_1 \,\middle|\, \ell \neq b\right\} \cup \left\{\left(\ell, \frac{n}{4} + \ell\right) \in A_1 \times B_2 \,\middle|\, \ell \neq a\right\}$$

Our 1st Streaming LB: Maximum Matching (5)

$$E_2 = \left\{\left(\frac{n}{4} + \ell, \ell\right) \in A_2 \times B_1 \,\middle|\, \ell \neq b\right\} \cup \left\{\left(\ell, \frac{n}{4} + \ell\right) \in A_1 \times B_2 \,\middle|\, \ell \neq a\right\}$$



Observation: G has a matching of size $\frac{n}{2} - 1$ if and only if $X[k] = 1$, otherwise G has a matching of size $\frac{n}{2} - 2$

Our 1st Streaming LB: Maximum Matching (6)

Proof. (continued)

- Bob continues the execution of $\boldsymbol{A}$ on E_2
- If the output is a matching of size $\frac{n}{2} - 1$ then Bob reports $X[k] = 1$, otherwise (i.e., the size is $\frac{n}{2} - 2$) Bob reports $X[k] = 0$.

□

Summary and Outlook

Summary:

- We introduced the one-way two-party communication model for deterministic protocols
- We showed that $\mathrm{D}(\mathrm{Index}_n) = n$.
- We gave a first space lower bound for deterministic streaming algorithms by a reduction to the Index communication problem

Outlook:

- Shortcoming: Lower bound only holds for deterministic algorithms!
- We'll look into randomized lower bounds in the next lecture