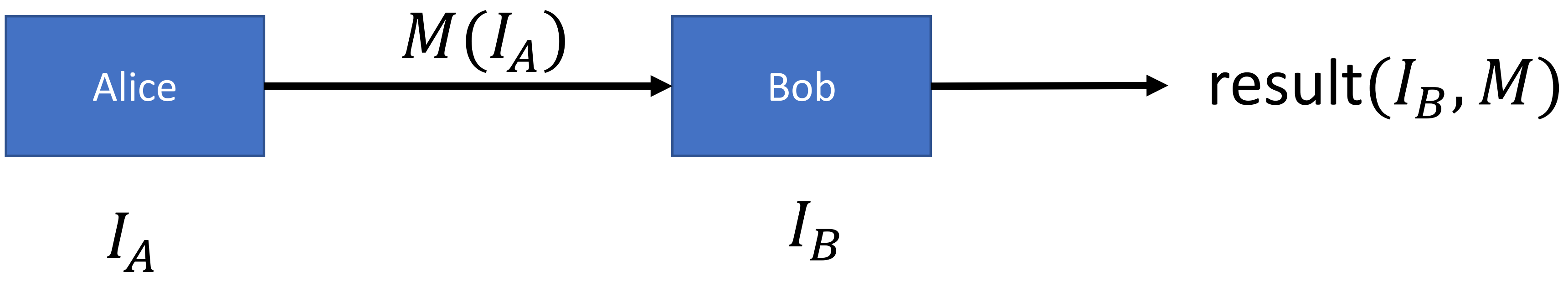
Advanced Topics in Theoretical Computer Science

Lecture 16

Randomized Communication Complexity

Deterministic Communication Complexity

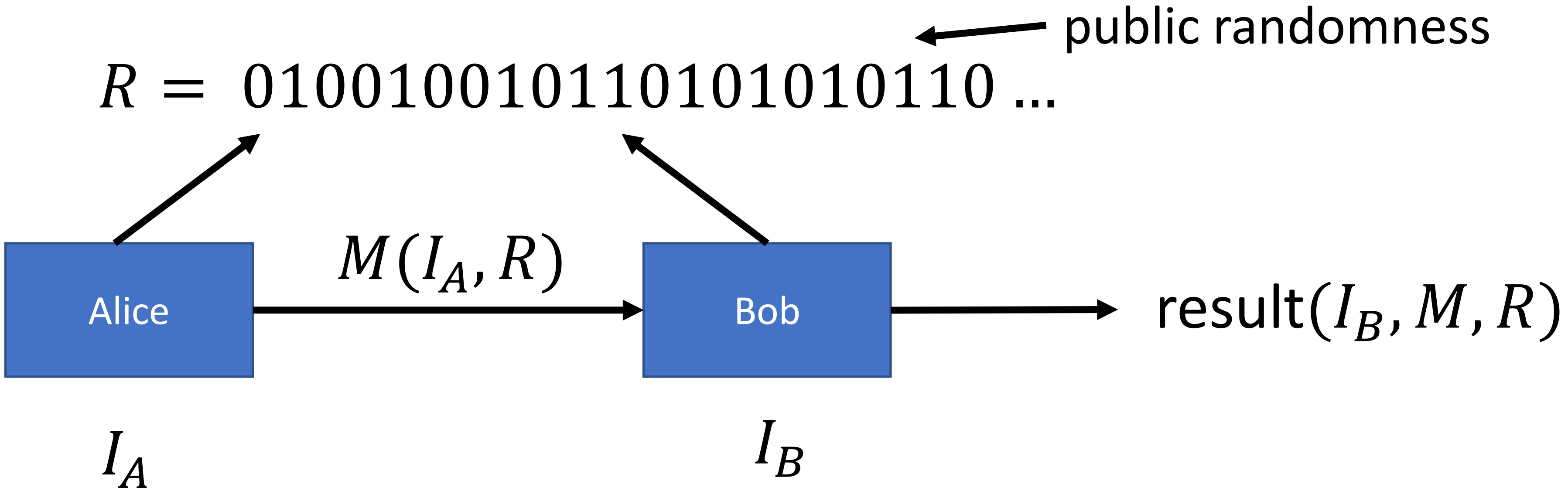
Deterministic Communication Complexity:



- M is a function of I_A
- Output is a function of M and I_B
- No randomization, same input → same output
- Protocol successful on all inputs

Randomized Communication Complexity

Randomized Communication Complexity:



- **Public Randomness:** Parties have access to a shared random bit-string
- Protocol needs to succeed with proba. $> \frac{1}{2}$ over the public randomness
- Randomized CC $R_\epsilon(f)$ is minimum cost over all randomized communication protocols that succeed with probability at least $1 - \epsilon$

Example: Equality

Equality:

- Alice holds $X \in \{0,1\}^n$, Bob holds $Y \in \{0,1\}^n$
- They wish to compute the equality function:

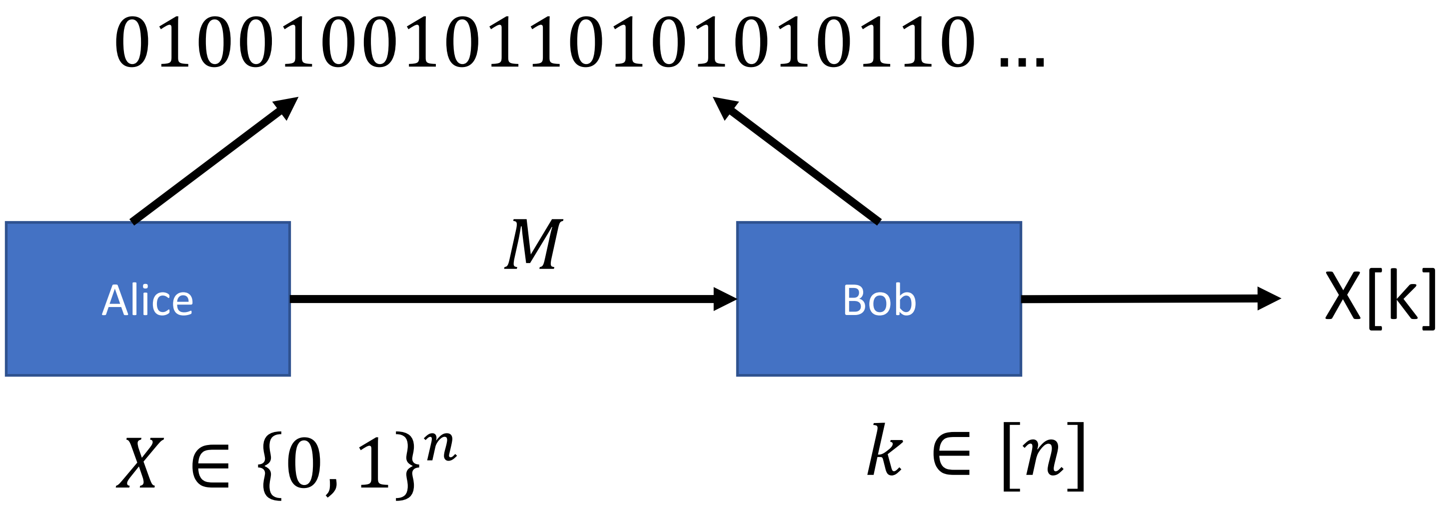
  $EQ(X,Y) = 1$, if $X = Y$, and $EQ(X,Y) = 0$ otherwise

Deterministic Communication Complexity: $D(EQ_n) = \mathrm{n}$ (exercise!)

Randomized Communication Complexity: $R_{0.99}(EQ_n) = \Theta(\log n)$

- Alice and Bob compute a hash function $h: \{0,1\}^n \to [\Theta(\log n)]$ using public randomness as seed for the hash function (see Adv. Alg.)
- It can be seen that protocol succeeds with probability at least 0.99.

Randomized CC of INDEX



Can we make use of the shared rand. bits to solve INDEX with message size $o(n)$?

Theorem. $R_{\frac{2}{3}}(\mathrm{INDEX}_n) = \Omega(n)$

→ Randomized one-pass streaming algorithms for Maximum Matching also require space $\Omega(\mathrm{n}^2)$. (see previous lecture)

Protocols that are Good on Average

Lower Bounds for Randomized Protocols:

- Proving lower bunds for randomized protocols directly is difficult
- Yao's Lemma allows us to consider deterministic protocols instead, albeit using a different *quality guarantee*...

Deterministic Protocols: For every input (I_A, I_B) the protocol succeeds

Deterministic Protocols that are Good on Average:
Let μ be an input distribution so that $(I_A, I_B) \sim \mu$. A deterministic protocol P computes a function f up to error δ with respect to μ if

$$\mathbb{P}_{(I_A, I_B) \sim \mu}[P(I_A, I_B) \text{ fails}] \leq \delta.$$

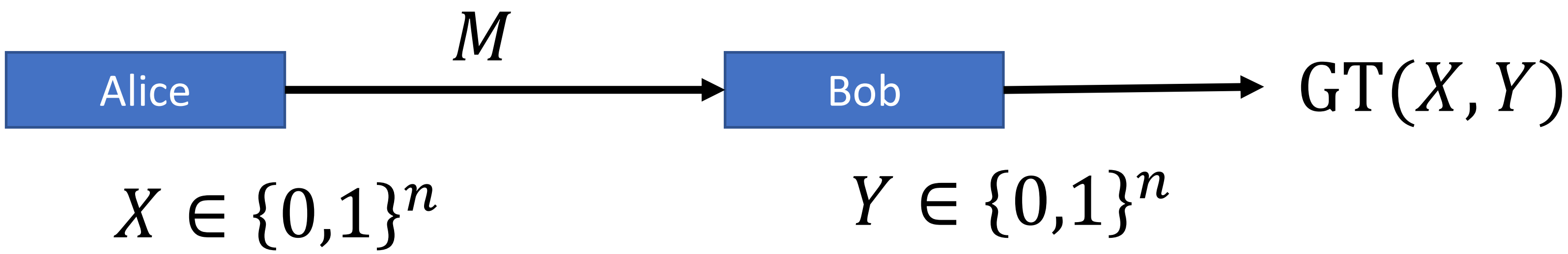
"Deterministic protocol with distributional error"

Distributional CC: δ-error μ-distributional deterministic communication complexity D_δ^μ is minimum cost of a protocol that satisfies previous inequality

Example: “Greater than” Function

Greater-than (GT) function:

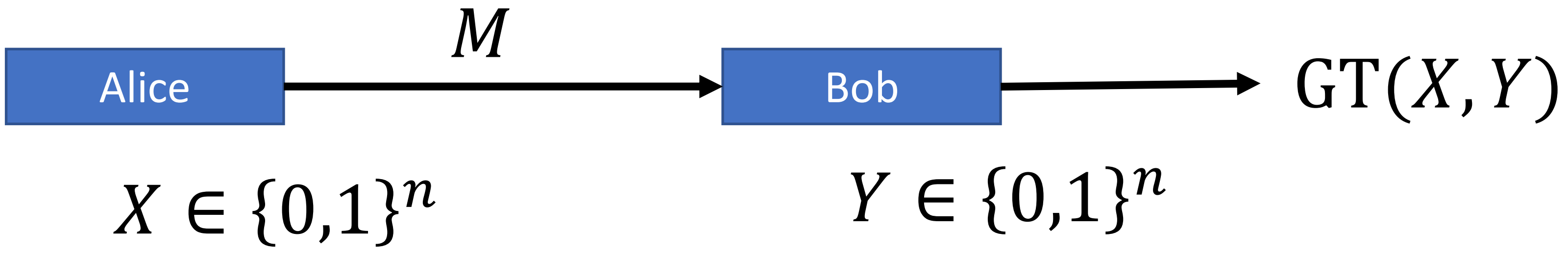
$$\mathrm{GT}(x, y) = \begin{cases} 1, \text{if } x \geq y \\ 0, \text{otherwise} \end{cases}.$$



X, Y are binary representations of numbers in $\{0, 1, \ldots, 2^n - 1\}$

Deterministic CC: $D(GT_n) = n.$

Example: "Greater than" Function



$\frac{1}{4}$-error Uniform Distributional CC:

- Consider the uniform dist., where X, Y are chosen independently from $\{0, 1\}^n$
- Consider the following protocol: Alice sends position of her most significant bit $MSB(X)$, i.e., position of left-most "1", and value $n + 1$ if $X = 0 \ldots 0$ ($\rightarrow$ message of size $\lceil \log(n + 1) \rceil$)
- Bob outputs "1" if $MSB(X) \leq MSB(Y)$

Example:

$$X = 0\,1\,0\,1 \rightarrow MSB(X) = 2$$
$$Y = 1\,0\,0\,1 \rightarrow MSB(Y) = 1$$

Output = 0 (correct)

Example: “Greater than” Function

$$
\begin{array}{ccc}
 & & \begin{array}{cccccccc} 4 & 3 & 2 & 2 & 1 & 1 & 1 & 1 \end{array} \\
 & Y & \\
X & & \begin{array}{cccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\
\begin{array}{c} 4 \\ 3 \\ 2 \\ 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{array} & \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} & \begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & \mathbf{0} & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
1 & 1 & 1 & 1 & 1 & 1 & \mathbf{0} & \mathbf{0} \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & \mathbf{0} \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}
\end{array}
$$

$n = 3$

Example: “Greater than” Function

$$
\begin{array}{cc|cccccccc}
 & & 4 & 3 & 2 & 2 & 1 & 1 & 1 & 1 \\
 & & & & & Y & & & & \\
X & & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
4 & 0 & \multicolumn{8}{c}{\multirow{8}{*}{$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \mathbf{0} & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & 1 & 1 & 1 & 1 & 1 & \mathbf{0} & \mathbf{0} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & \mathbf{0} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$}} \\
3 & 1 \\
2 & 2 \\
2 & 3 \\
1 & 4 \\
1 & 5 \\
1 & 6 \\
1 & 7
\end{array}
$$

$n = 3$

Theorem.

$\mathbb{P}[\text{correct}] \geq \frac{3}{4}.$

$\rightarrow$

$D^{uni}_{\frac{1}{4}}(GT_n)$

$\leq \lceil \log(n+1) \rceil$

Yao's Lemma

Connecting Randomized CC to Deterministic CC with distributional Error

Yao's Lemma.

$$R_\delta(f) = \max_\mu D_\delta^\mu(f),$$

where max. is taken over all prob. distributions μ on the domain of f.

Recall:

- **Randomized CC:** probability of error $\mathrm{at\ most}$ δ on any input instance (randomness in random bits)
- **Deterministic CC with distributional Error:** probability of error at most δ (randomness over input distribution)

Randomized CC of INDEX

Theorem. $\mathrm{R}_{\frac{1}{10}}(\mathrm{INDEX_n}) = \Omega(\mathrm{n})$

Proof.

- Let P be a randomized protocol for INDEX_n with cost $\mathrm{cn} = \mathrm{R}_{\frac{1}{10}}(\mathrm{INDEX_n})$ and error probability $\frac{1}{10}$, for some $c < \frac{1}{10}$
- We will prove that such a protocol does not exist in the following
- By Yao's Lemma, there exists a deterministic protocol Q for INDEX_n with $cost(Q) \leq cn$ and dist. error proba. $\frac{1}{10}$ over the uniform distribution (i.e., $X \in \{0,1\}^n$ and $k \in [n]$ are chosen uniformly at random)
- Since $cost(Q) = cn$, Q sends at most 2^{cn} different messages $M_1, M_2, \ldots, M_{2^{cn}}$. Denote by $M(x)$ the message sent by Alice when Alice holds input x.
- Denote by $out(M_i, k)$ the output produced by Bob when Bob's input is k and M_i is received. Let $out(M_i) = \big(out(M_i, 1), out(M_i, 2), \ldots, out(M_i, n)\big)$ be the output vector of length n

Randomized CC of INDEX

Proof. (continued)

- Denote by M_i^{-1} the set of Alice's inputs on which she sends message M_i to Bob
- Let $x \in M_i^{-1}$. Then:

$$\mathbb{P}[\mathrm{error} \mid \mathrm{X} = \mathrm{x}] = \frac{\mathrm{d_H}\big(x, out(M_i)\big)}{n},$$

where $d_H(a, b)$ denotes the Hamming distance (number of positions where the two strings are not equal) between a and b.

- We thus obtain:

$$\mathbb{P}[error] = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \mathbb{P}[error \mid X = x] = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \frac{\mathrm{d_H}\big(x, out(M(x))\big)}{n}.$$

Randomized CC of INDEX

Proof. (continued)

- Let $out = \{out(M_i) \mid i \in [2^{cn}]\}$ be the set of all output vectors
- Let X_{good}, X_{bad} be a partition of the set of Alice's inputs $\{0,1\}^n$, s.t.

$$X_{good} \coloneqq \{\mathrm{x} \in \{0,1\}^n : \exists i \text{ such that } d_H\big(x, out(M_i)\big) \le \frac{n}{4}\}$$
$$X_{bad} \coloneqq \{0,1\}^n \setminus X_{good}$$

- **Observe:** $\mathbb{P}[error \mid x \in X_{bad}] \ge 1/4$
- We will show now that X_{good} is small and most of Alice's inputs are in X_{bad}
- This then implies that the average error is large!

Randomized CC of INDEX

Proof. (continued)

- **Claim.** $\left|X_{good}\right| \leq 2^{0.961n+o(n)}$.
- **Proof of Claim.**
    - Consider $out(M_i)$, for any i. Then:

$$|\{y \in \{0,1\}^n : d_H(out(M_i), y) \leq \frac{n}{4}| = \sum_{0 \leq i \leq \frac{n}{4}} \binom{n}{i} \leq \sum_{0 \leq i \leq \frac{n}{4}} \left(\frac{en}{i}\right)^i \leq \frac{n}{4} \cdot \left(\frac{en}{\frac{n}{4}}\right)^{\frac{n}{4}}$$

$$\leq n\,(4e)^{\frac{n}{4}}$$

$$n\,(4e)^{\frac{n}{4}} = n2^{\log_2(4e)n/4} \leq n\,2^{0.861n} = 2^{0.861n+\log n}$$

    - Hence:

$$\left|X_{good}\right| \leq 2^{cn} \cdot\ 2^{0.861n+\log n} = 2^{cn+0.861n+o(n)} \leq 2^{0.1n+0.861n+o(n)}$$

$\square$

Randomized CC of INDEX

Proof. (continued)

- We thus obtain:

$$\mathbb{P}[error] = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \mathbb{P}[error \mid X = x] = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \frac{\mathrm{d_H}\big(x, out(M(x))\big)}{n}$$

$$\geq \frac{1}{2^n} \sum_{x \in X_{bad}} \frac{\frac{n}{4}}{n} = \frac{1}{4} \cdot \frac{1}{2^n} \cdot |X_{bad}| \geq \frac{1}{4} \cdot \frac{1}{2^n} \cdot (2^n - 2^{0.961n + o(n)}) = 1/4 - o(1).$$

- This contradicts the error probability of $\frac{1}{10}$. Hence, protocols P and Q cannot exist!

$\square$

Lower Bound for CONNECTIVITY

Theorem. Every one-pass randomized streaming algorithm with error probability at most 0.1 for deciding CONNECTIVITY requires space $\Omega(n)$.

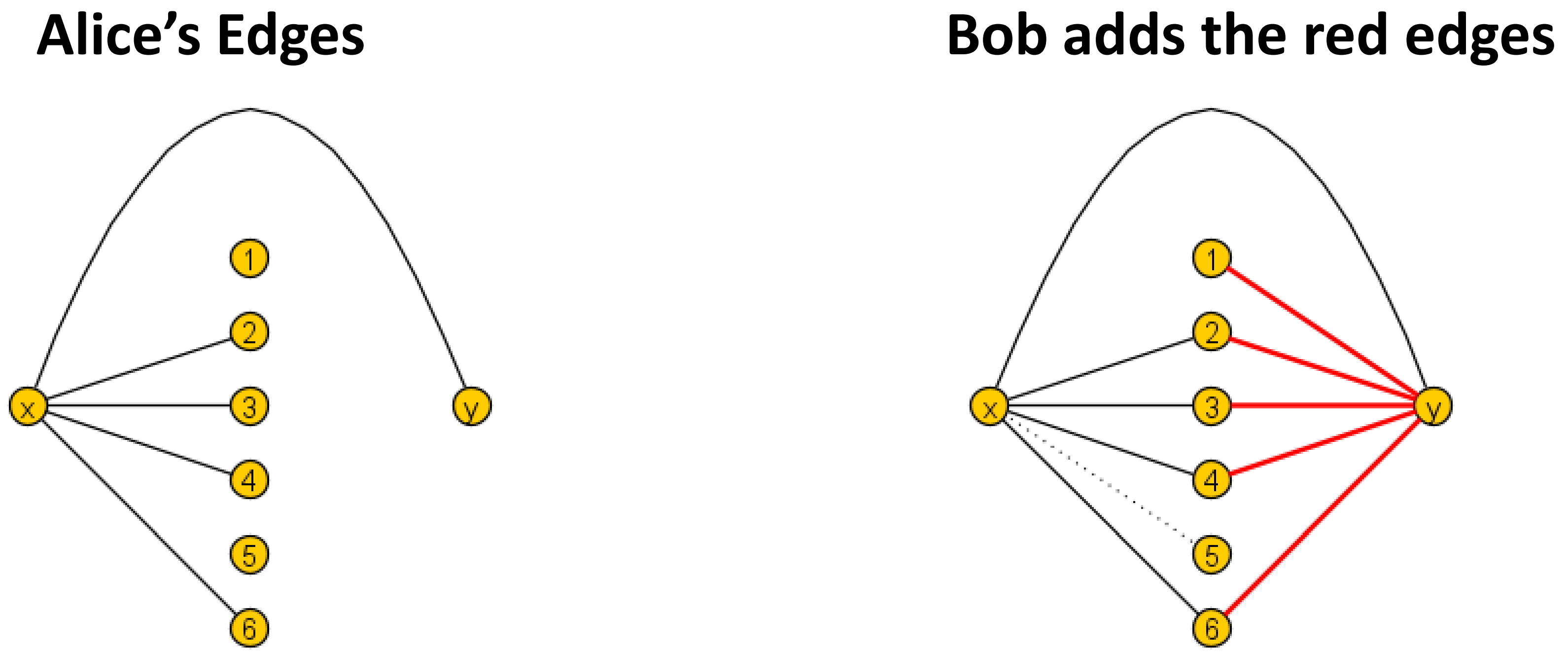
Proof.

- Let A be a one-pass randomized streaming algorithm for CONNECTIVITY.
- We will show how A can be used to solve INDEX_{n-2}. Since $R_{0.1}(\mathrm{INDEX}_{n-2}) = \Omega(n)$, the result follows.
- Consider thus an instance $(X, k) \in \{0,1\}^{n-2} \times [n-2]$ of INDEX_{n-2}
- Given X, Alice's constructs the following graph $G_1 = (V \cup \{x, y\}, E_1 \cup \{(x, y)\})$, with $V = [n-2]$ and $(i, x) \in E_1 \Leftrightarrow X[i] = 1$
- Alice runs algorithm A on $E_1 \cup \{(x, y)\}$ and sends state of algorithm to Bob
- Bob adds edges $E_2 = \{(i, y) \mid 1 \leq i \leq n-2 \text{ and } i \neq k\}$ and completes the algorithm
- Bob outputs $X[k] = 1$ if graph connected and $X[k] = 0$ if graph not connected. $\square$

Lower Bound for CONNECTIVITY (2)

Example (n=8).

- Alice holds $X = 0\ 1\ 1\ 1\ 0\ 1 \in \{0,1\}^{n-2} = \{0,1\}^6$
- Bob holds index $k = 5$



Graph connected $\Leftrightarrow \boldsymbol{X[k] = 1}$

Summary

Summary:

- We have proved that the one-way two-party randomized CC of INDEX_n is $\Omega(n)$
- To this end, we considered deterministic protocols with distributional error and applied Yao's lemma

Streaming Applications:

- We have already seen that a streaming algorithm for MAXIMUM MATCHING can be used to solve $\mathrm{INDEX}_{\Theta(n^2)}$
- Since we now know that not only $D(\mathrm{INDEX}_n) = \Omega(n)$ but also $R_{\frac{1}{3}}(\mathrm{INDEX}_n) = \Omega(n)$, every randomized streaming alg. for MAXIMUM MATCHING requires space $\Omega(n^2)$
- We have also seen that solving CONNECTIVITY requires $\Omega(n)$ space for every randomized streaming algorithm