Umans Complexity Theory Lectures

Communication Complexity: Deterministic and Randomized

(with additional material from Maverick Woo Lecture)

Communication Complexity

- Goal: Isolate computational bottlenecks of certain problems and provide tools for resolving their complexity

- First results: Yao, Some Complexity Questions Related To Distributed Computing, STOC 1979

Communication Complexity

Yao's Model for Distributed Communication:

- Only two distant parties, say Alice and Bob.
- Both parties cooperate.
- Both parties know beforehand what the computation problem is.
- Each party get a fixed part of the input.
- Only concerned about communication, not computational resources like time or space.

Communication Complexity

Two Communicating parties:

Alice and Bob

Wish to Jointly Compute Function

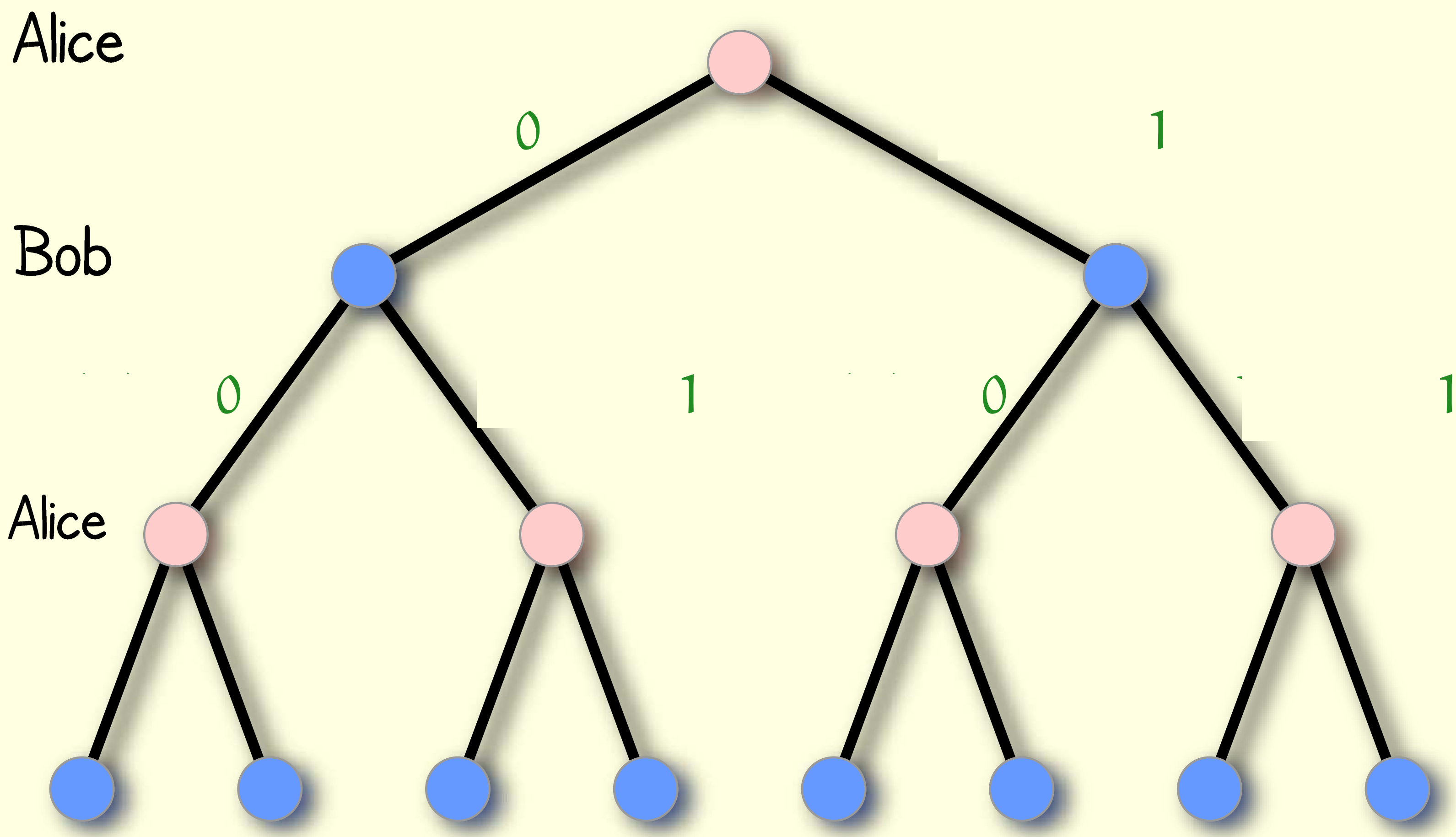
$f:\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$

Alice holds $x \in \{0,1\}^n$; Bob holds $y \in \{0,1\}^n$

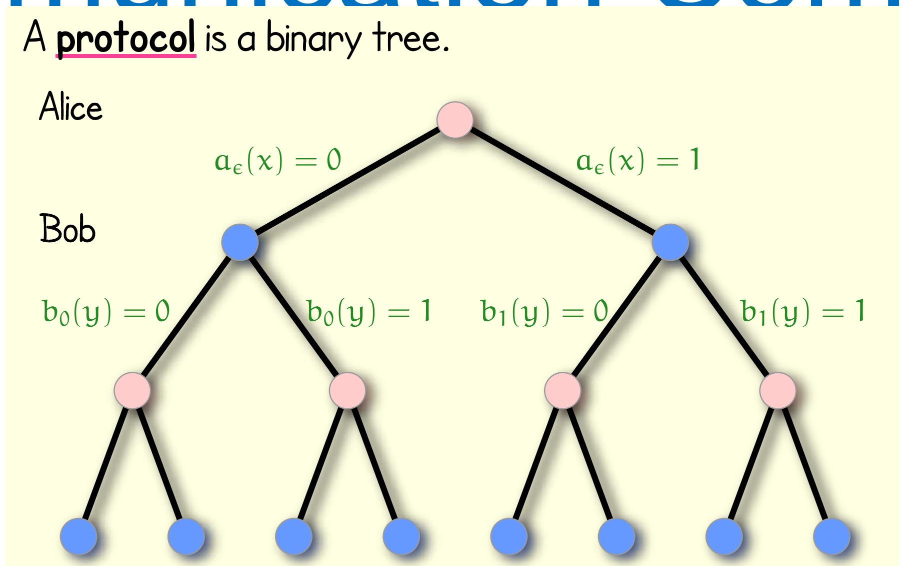
- Goal: **collaboratively** and **distributively** compute $f(x, y)$ while communicating as few bits as possible between Alice and Bob
- Count number of bits exchanged (computation free)
- At each step: one party sends bits that are a function of held input and received bits so far

Communication Complexity

A **protocol** is a binary tree.



Communication Complexity



- Each leaf is labelled either 0 or 1.
- A protocol is correct if for any x, y, the protocol arrives at a leaf labelled f(x, y).
- The cost of the protocol is the length of its longest path: the max number of bit communications between the parties
- Note: Internal computation at each side is free.

Communication Complexity

Note:

- It is not necessary for the two parties to strictly alternate.
- The tree is full but not necessarily complete.
- Different input pairs can arrive at the same leaf (thus via the same path).

Deterministic Communication Complexity

- Example function (equality):

Suppose $f(x, y) = EQ(x, y)$ where

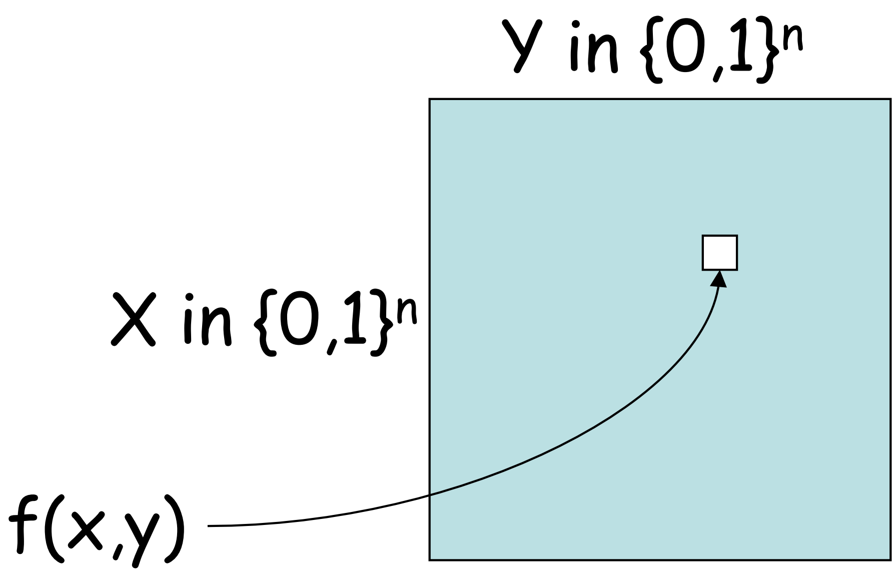
$EQ(x, y) = 1$ if $x = y$, and otherwise $EQ(x, y) = 0$

- Communication protocol for $EQ(x, y)$:
  - Alice sends x to Bob (n bits)
  - Bob sends $EQ(x, y)$ to Alice (1 bit)
  - Total: $n + 1$ bits communicated
  - (works for any predicate f with 0,1 output)

Deterministic Communication Complexity

<u>Theorem</u>: No deterministic protocol can compute EQ(x, y) while exchanging fewer than n+1 bits.

- Proof:
  - "input matrix":



Deterministic Communication Complexity

Mono-chromatic Rectangles:

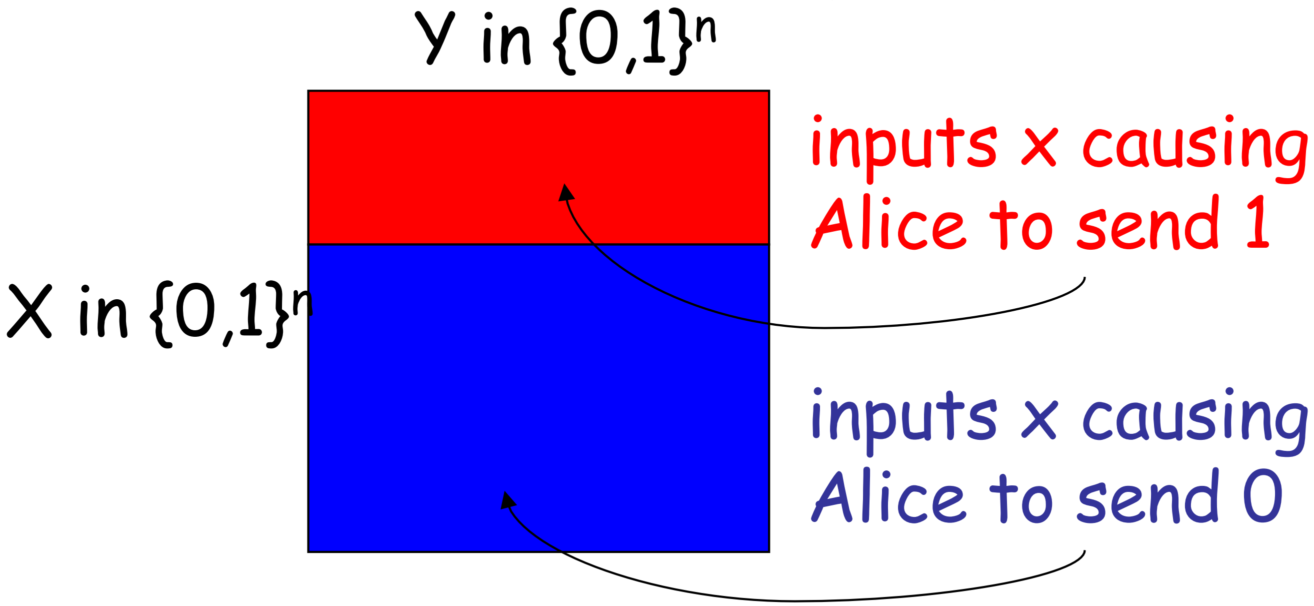
- If X and Y are subsets of $\{0, 1\}^n$, then $X \times Y$ is called a mono-chromatic rectangle if:

f(x,y) is not distinguishable by the parties within the rectangle.

- Each node in a protocol splits a mono-chromatic rectangle into two.

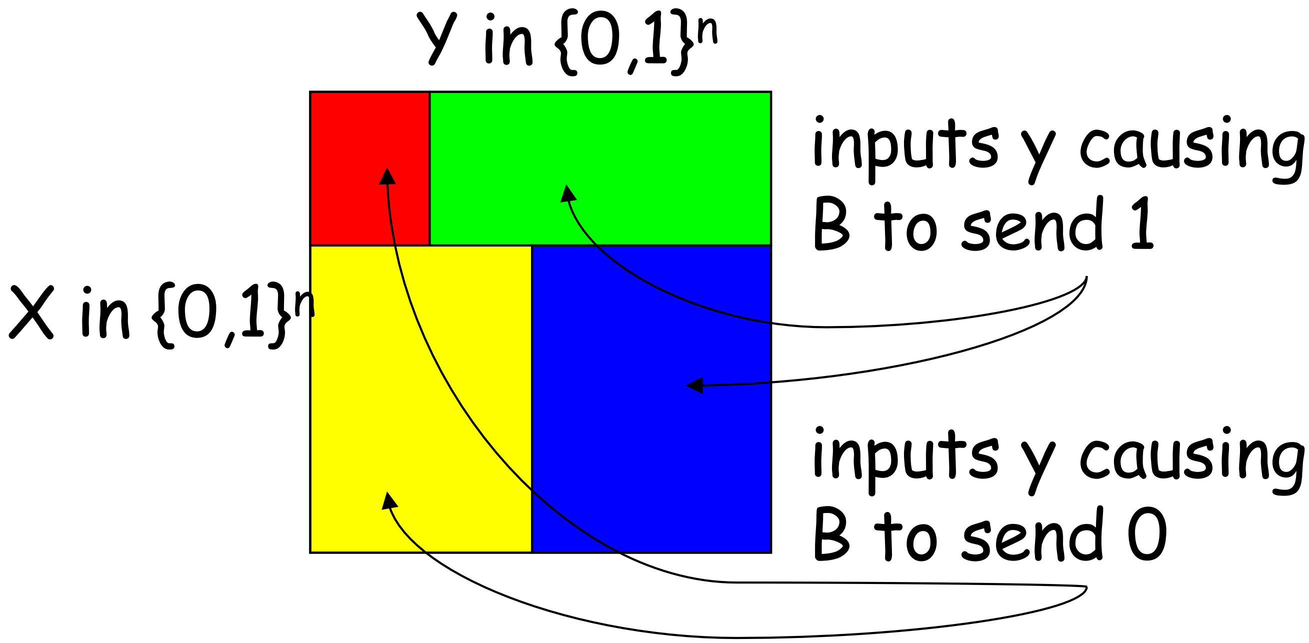
Deterministic Communication Complexity

- Communication between Alice and Bob
- Assume 1 bit sent at a time, alternating (same proof works in general setting)
- Alice sends 1 bit depending only on x:



Deterministic Communication Complexity

– Bob sends 1 bit depending only on y and received bit:



Deterministic Communication Complexity

– At end of protocol involving k bits of communication, matrix is partitioned into at most 2^k **mono-chromatic rectangles**

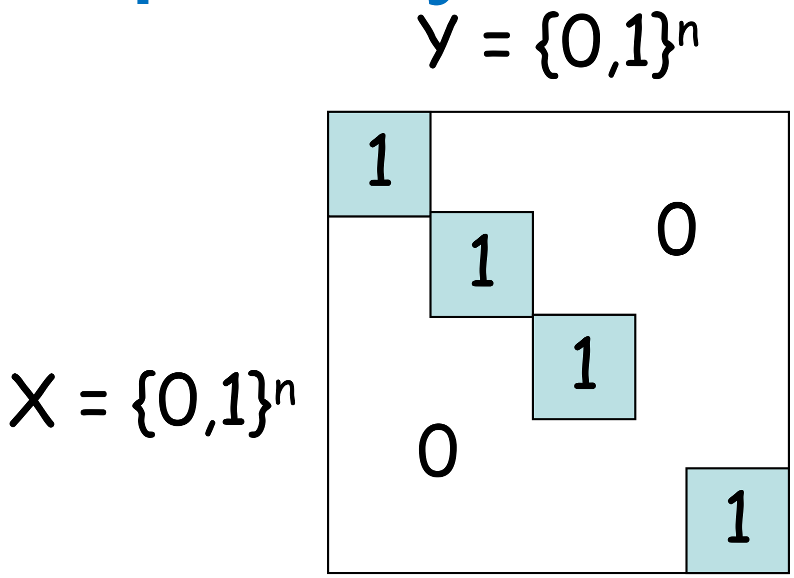
– Bits sent in protocol are the same for every input (x, y) in given rectangle

– Conclude: **At end of protocol:**

f(x,y) must be constant on each mono-chromatic rectangle

Deterministic Communication Complexity

Matrix M for EQ:



- Any partition into mono-chromatic rectangles with constant f(x,y) must have **$2^n + 1$ mono-chromatic rectangles**.
- Any EQ Protocol that exchanges ≤ n bits can only create 2^n rectangles, so **any EQ Protocol must exchange at least n+1 bits.**

Deterministic Communication Complexity

Using "Fooling Set Argument" for Lower Bounds:

- Consider (x_1, y_1) and (x_2, y_2)
- Suppose f has the same value on them, say z.
- If we can show that the value of f is not z in at least one of (x_1, y_2) or (x_2, y_1) then (x_1, y_1) and (x_2, y_2) cannot possibly be in the same leaf.
- Hence a lower bound on the number of leaves

Deterministic Communication Complexity

Copy and Paste Lower Bound Argument using Fooling Set Argument:

- It is crucial that Alice does not know y and Bob does not know x.
- Copy: the way Alice behaves for the communication on (x_1, y_1) and the way Bob behaves for the communication on (x_2, y_2)
- Paste: interleave together these communications to get the communication of (x_1, y_2), where they both behave the same.

Using Fooling Set Argument for Lower Bounds of EQ:

- **EQ has fooling set of size 2^n:**

  $$S = \{(x, y) | x=y \text{ and } x,y \in \{0, 1\}^n\},$$

since each party behaves the same only if x=y.

- So we have at least 2^n leaves labeled with 1.
- We still need to have a leaf labeled with 0, thus we need at least n + 1 bits to distinguish them all.
- Since we already know an (n + 1)-bit solution, this bound is tight for EQ.

Deterministic Communication Complexity

Another method for Lower Bounds of Deterministic Communication: The Rank Method

– Let M be the 0-1 matrix of size 2^n x 2^n where $M_{x,y} = f(x, y)$

– Each leaf of the protocol is associated with a rectangle with only 0´s or only 1´s.

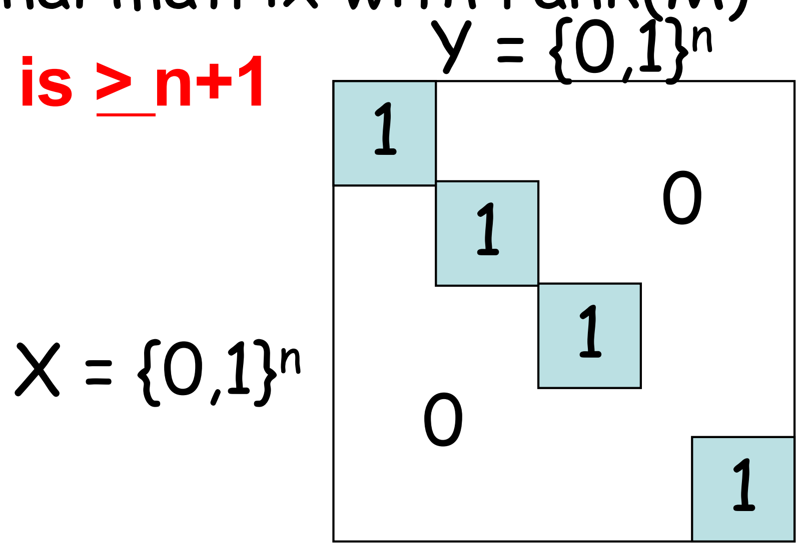
– Can show **Rank(M) $\leq$ number of leaves of protocol**

– Since a protocol is a binary tree, its communication cost is at least 1+log(leaves) $\geq$ 1+log(rank(M)),

- *Matrix M for EQ is diagonal matrix with rank(M)=* 2^n*, so*

communication cost of EQ is $\geq$ n+1

Matrix M for EQ:



Randomized Communication Complexity

- Can we do better than a deterministic protocol?
  - Probabilistic protocol:
    - At each step: one party sends bits that are a function of held input and received bits so far and the result of some coin tosses
    - Required to output f(x, y) with high probability over all coin tosses

Randomized Communication Complexity

- Protocol for EQ employing randomness:
  - Alice picks random prime p in $\{1...4n^2\}$, sends:
    - **p**
    - **(x mod p)**
  - Bob sends:
    - **(y mod p)**

  Players output 1 if and only if:

  $(x \bmod p) = (y \bmod p)$

  And otherwise players output 0.

Randomized Communication Complexity

Randomized Protocol with O(log n) bits exchanged:

– Case $x = y$: always correct

– Case $x \neq y$: incorrect if and only if:

p divides $|x - y|$

Claim: probability incorrect $\leq$ 1/2

– # primes in range {1...N} is $\geq$ sqrt(N), so

– # primes in range $\{1...4n^2\}$ is $\geq 2n$

– # primes dividing $|x - y|$ is $\leq n$

– probability incorrect $\leq$ ½

(Can repeat to get high probability of correctness)

=> Randomness gives an exponential advantage!!