BPP lifting using Inner product

Sajin Koroth
(Simon Fraser University)

Joint work with

Arkadev Chattopadhyay
Yuval Filmus
Or Meir
Toniann Pitassi

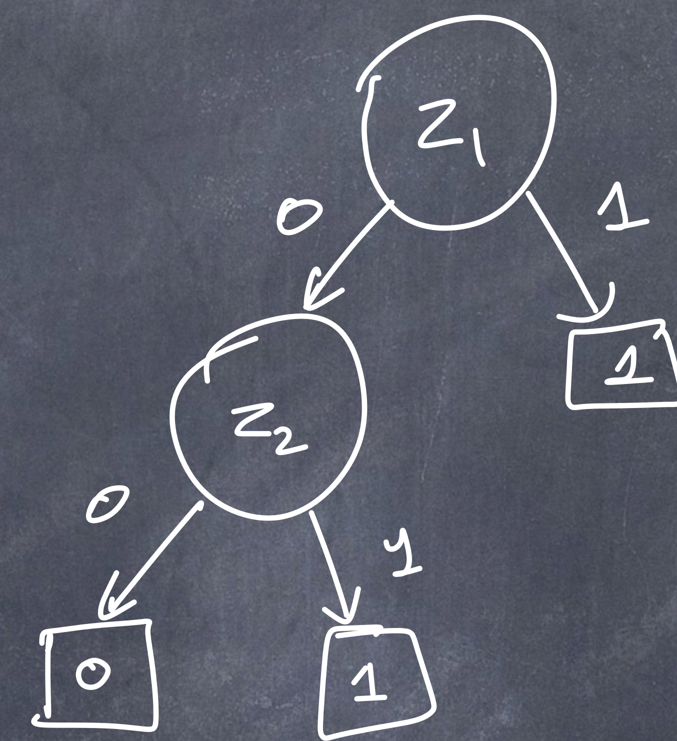



Lifting

- Transfer lower bounds from weak models to strong
- Weak : Decision Trees
- Strong : Communication complexity

Decision Trees

- Model : Tree
  Querying variables
- Cost : Depth
- Weak model ->
  Easy to prove lower bounds

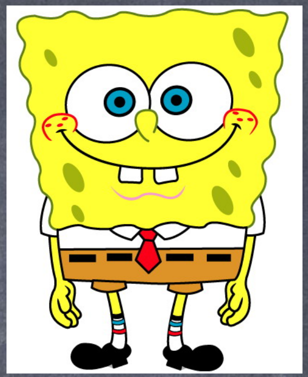


Decision tree computing

2-bit OR

Communication Complexity

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

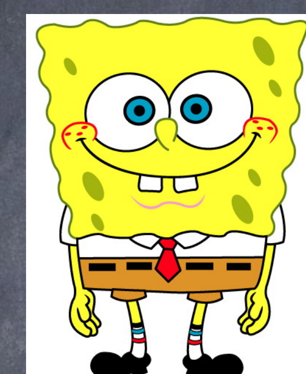


$X \in \{0,1\}^n$ $\qquad$ $Y \in \{0,1\}^n$

compute $f(X, Y)$

Cost : number of bits communicated

Communication Complexity

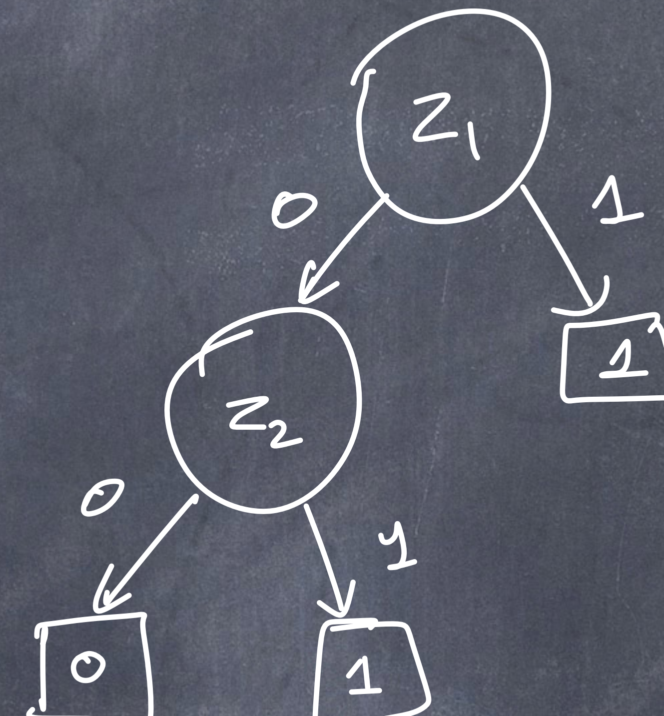


$X \in \{0,1\}^n$ $\qquad$ $Y \in \{0,1\}^n$

Is there an index $i \in [n], X_i = 1$ and $Y_i = 1$?

$DISJ_n(X, Y)$ asks whether if $X \cap Y = \phi$

Tying it all: Query to Communication

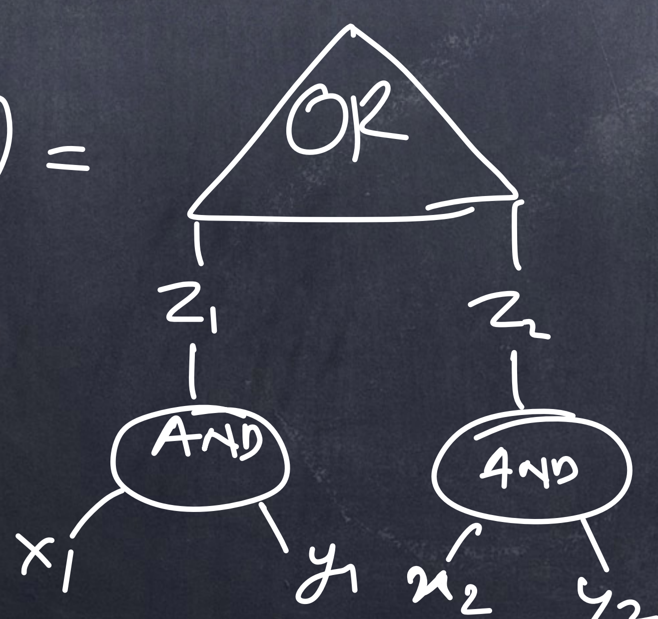


$\underline{\mathrm{DISJ}_n}$

Alice	Bob
$X = x_1, x_2 \in \{0,1\}^2$	$Y = y_1, y_2 \in \{0,1\}^2$

Goal: Compute if $X \cap Y = \emptyset$?

$\mathrm{DISJ}_n(X, Y) =$

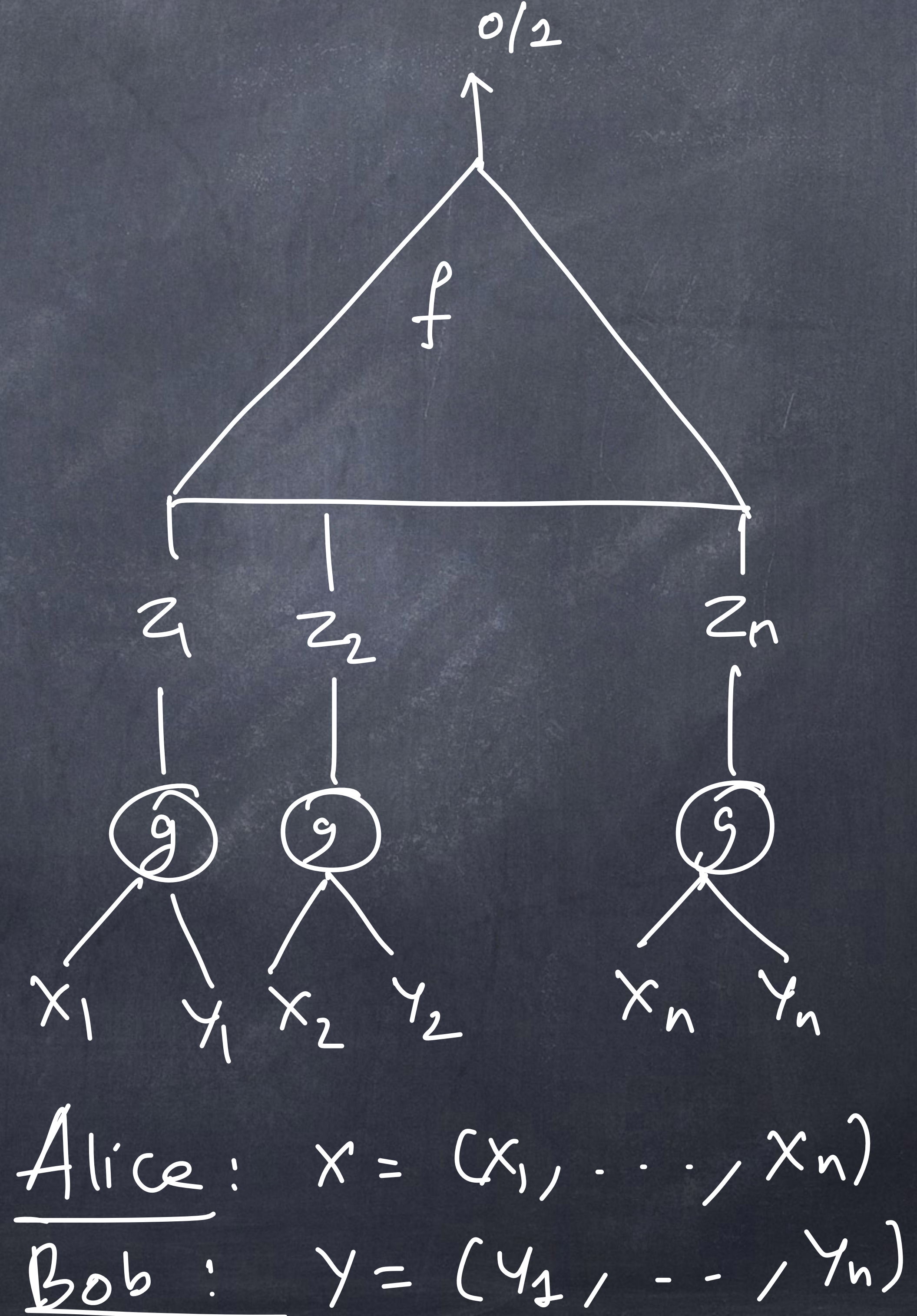


Decision tree computing

2-bit OR

Lifting to Communication Complexity

- One party function:
  $f : \{0,1\}^n \to \{0,1\}$
- Two party gadget :
  $g : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$
- Composed 2 party function: $f \circ g$



Lifting theorems
Meta statement

Decision Tree Complexity

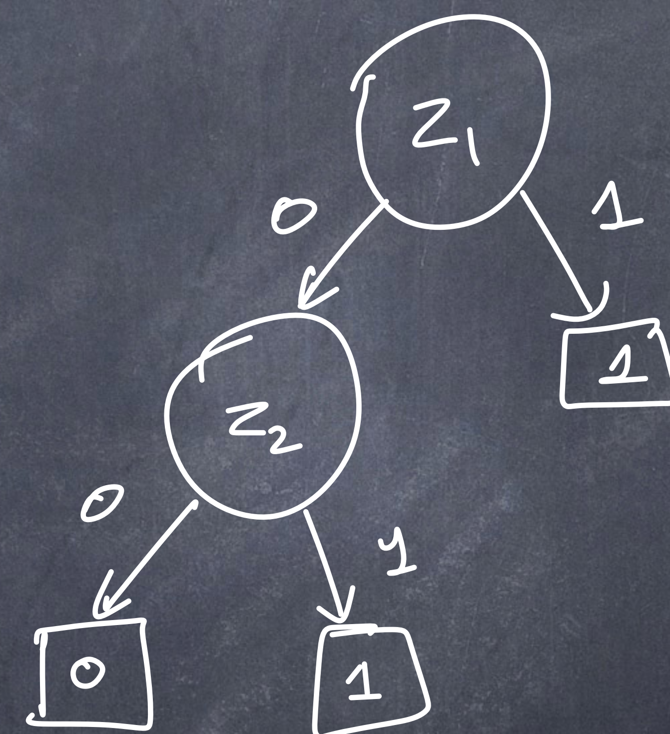
$$CC(f \circ g) \approx Dt(f) \times CC(g)$$

Communication complexity

Easy direction of Lifting

$$CC(f \circ g) \leq Dt(f) \times CC(g)$$

$f \circ g$	Alice	Bob
	x_1, x_2	y_1, y_2



$f = OR$

Protocol for $f \circ g$

(1) Alice and Bob compute $z_1 = g(x_1, y_1)$ using best protocol for g.

(2) If $z_1 = 1$ o/p 1 and stop.
Else compute z_1

Easy direction of lifting

$$CC(f \circ g) \leq Dt(f) \times CC(g)$$

- T : best decision tree for f
- Protocol :
  - Alice and Bob simulates T
  - When T queries z_i , they compute $g(x_i, y_i)$
  - Cost : #queries x CC(g)

Why Lifting theorems?

- A fundamental question about computation
  - Is Naive = Optimal?
  - Direct sum
  - XOR Lemmas
- Separations of complexity classes

Brief survey of results

- [Ran Raz, Pierre McKenzie '97, G¨o¨os, Pitassi, Watson '15]
  - Deterministic Decision tree -> Deterministic communication
  - $P^{CC}(f \circ INDEX) = P^{Dt}(f) \times \Theta(\log n)$

$\underline{INDEX_n}$

$\underline{\text{Alice}}$

$x \in [n]$

$\#bits(x) = \log n$

$\underline{\text{Bob}}$

$$y = \begin{array}{|c|c|c|c|c|} \hline 0 & 1 & 0 & & 0 \\ \hline \end{array}$$
$$\begin{array}{ccccc} 1 & 2 & 3 & \cdots & n \end{array}$$

$\#bits(y) = n$

$\underline{\text{Goal:}}$ Find y_x

Brief survey of results

- [Chattopadhyay, Koucký, Loff, Mukhopadhyay '17, Wu, Yao, Yuen'17]
  $P^{CC}(f \circ IP) = P^{Dt}(f) \times \Theta(\log n)$

$IP:$ Alice Bob. Goal : Compute

$X \in \{0,1\}^{\log n}$ $Y \in \{0,1\}^{\log n}$ $< X, Y > = \oplus_i (X_i \wedge Y_i)$

Brief survey of results

- [G¨o¨os, Pitassi, Watson '17]
  - Randomized Decision tree -> Randomized communication
  - $BPP^{CC}(f \circ INDEX) = BPP^{Dt}(f) \times \Theta(\log n)$

Story so far

- [GPW'15]
  $P^{CC}(f \circ INDEX) = P^{Dt}(f) \times \Theta(\log n)$
- [CKLM'17, WYY'17]
  $P^{CC}(f \circ IP) = P^{Dt}(f) \times \Theta(\log n)$
- [GPW'17]
  $BPP^{CC}(f \circ INDEX) = BPP^{Dt}(f) \times \Theta(\log n)$

Our Result

Works even for search problems

$$BPP^{CC}(f \circ IP) = BPP^{Dt}(f) \times \Theta(\log n)$$

Logarithmic sized gadget of exponentially small discrepancy

*Also: deterministic lifting

Impliations

- Unification : Gives a unified proof of both deterministic and randomized lifting
- Improved gadget size
- Gives a simplified proof of GJ'16 : BPP lower bound for AND-OR, MAJORITY trees

Open Problems

- Smaller than logarithmic sized gadgets?
- "Low" discrepancy gadgets?

Questions ?

Thank You