Instance of communication problem

- Tree of depth n;
- Alice gets one outgoing edge for every vertex at odd depth
- Bob .... even depth
- Goal: Compute leaf

For simplicity Alice & Bob alternate.
Noisy channels:
- Channel takes symbols from finite alphabet;
- Transmits most as is;
- Flips (adversarially) \( \in \) fraction.

Interactive Coding Scheme

Alice's Edges \( \rightarrow \) Tree for Alice
Bob's Edges \( \rightarrow \) Tree for Bob
The Cola

Computationally Inefficient

- Schulman: '92, '93
  \downarrow
  - Braverman-Kalai
    \downarrow
    - Kol-Raz

- Haempler

- Gha Hari-Haempler

Low Error State of the Art

High Error SotA
Tree Codes

Solution tool due to [Shulman].

"Online Error Correcting Code"

\[
T : \Sigma^n \rightarrow \Gamma^n
\]

\[
\text{(Rate} \quad \frac{\log_2 |\Sigma|}{\log_2 |\Gamma|})
\]

\[i\text{'th symbol of } T(x_1, \ldots, x_n) \text{ only depends on } x_{i-1} \ldots x_i.\]

\[
\{s_3, s_4, \ldots\} \quad \text{Tree Encoder.}
\]
Distance Property?

\[ \Delta(T(0000000), T(0000001)) \leq \frac{1}{n}. \]

Can expect if

\[ \Delta(T(x), y) < \epsilon \cdot n \]

then \( x_1 \) is uniquely determined

More generally if

\[ \Delta(T(x)_{j \ldots n}, y_{j \ldots n}) < \epsilon \cdot (n-j) \]

\( \forall i \leq j \), then \( x_i \) is uniquely determined

Formalizing above gives defn. of \( \epsilon \)-free code.
Many Trees!

- Problem: Tree ①
- Alice Encoding: Tree ②
- Bob’s Encoding: Tree ③
- Tool: Tree code ④

- All are distinct entities.
- In particular, ① & ④ mentioned often...
Central Idea:

- Alice & Bob maintain & communicate states on problem tree
- If states agree proceed to next level
- If not backtrack
- Incrementally: Next state one of five given current state

⇒ Wish to communicate sequence of 5-ary symbols
⇒ Use tree code with $k = 5$
- Quantities to track (on problem tree)

- Common depth

- Separation Bob separation

-Invariant: At time $t$
  
  $\alpha.$ Common depth + $\beta.$ #errors $\geq t$

- Will skip details

Caveats: Tree code exists. But not constructive.
- Key Idea: Alice and Bob exchange sets of edges

A → B: \( S_1 \subseteq S_2 \subseteq S_3 \ldots \subseteq S_k \ldots \)

B → A: \( T_1 \subseteq T_2 \subseteq T_3 \ldots \subseteq T_k \ldots \)

- Alice: Maintain \( S_i \), subset of her edges that she wishes to tell Bob about

- Decides Bob’s communication to determine \( \sim T_i \) (set he seems to be sending)

- Uses \( S_i \cup \sim T_i \) to determine \( S_{i+1} \)
Analysis:

1. Encoded property $S_i | S_{i-1}$ is $O(1)$ bits.

2. $j_i = \text{time i\textsuperscript{th} relevant edge appears in } S_i \cup T_j$

Then # errors in $(\hat{j}_i - j_{i-1})$

= $O(j_i - j_{i-1})$
Efficiency + Brakerski - Kalai

- Path solutions so far need tree codes. ⇒ Non-constructive.
- Overcome this by a more "dynamic" status check.

( Brakerski - Kalai ) + ( Itaenpler )

Suppose initial problem has

A → B → log n bits per round

B → A → log n bits per round.
Solution:

- No coding / Simple ECC within round
- After each round of interaction
  
  $A \rightarrow B : h, h(s)$
  
  $B \rightarrow A : h', h'(s')$

- If any party finds hash mismatch
  it backtracks.

- Yields $\Omega(1)$ rate & $\Omega(1)$ error.

Why?

- $|h|, |h(s)| = O(\log n)$

- Round has few errors

  $\implies$ common length - (divergence) $\uparrow$

  Careful analysis: $E$-error $\Rightarrow 1 - \sqrt{E}$ - rate.
Randomized Eff vs. Det. Exp. Time?

- Can fix sequence of $h$'s in advance.
- Beats any fixed oblivious adversary with $\exp(-n)$ prob.
- Union bound $\Rightarrow$ beats all adversaries.

Hashing dominates Tree hashing... (for low error). (log$n$ bit rounds).