CODING FOR ERRORS IN INTERACTIVE COMMUNICATION

Jason MacDonald

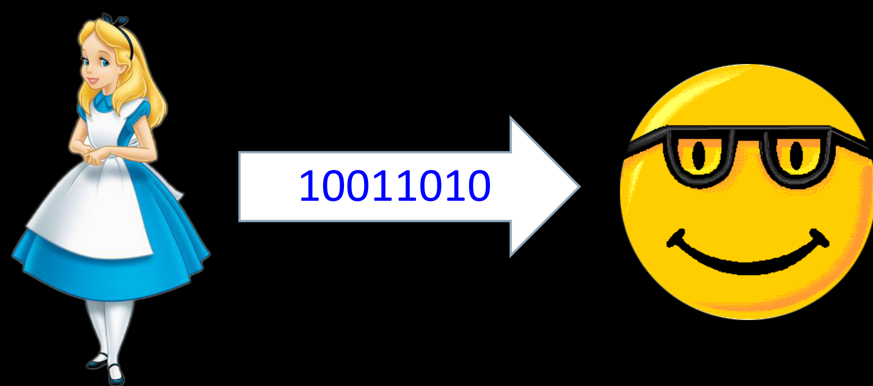
OVERVIEW

- Background
- Definitions
- Protocol
- Analysis

BACKGROUND

BACKGROUND

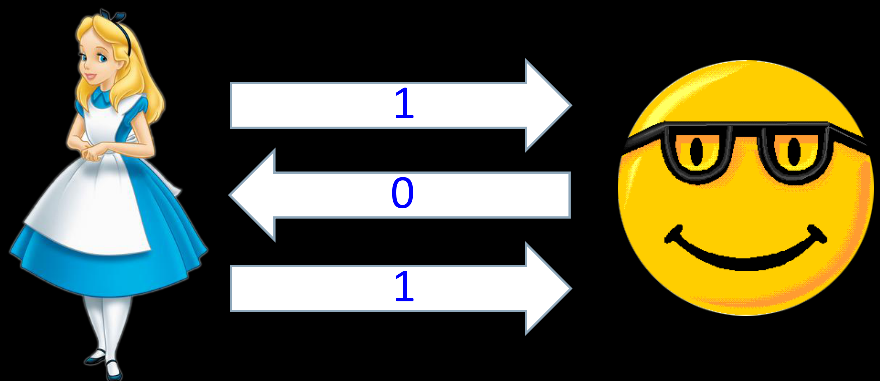
- Alice wants to send an *n*-bit message to Bob, but is worried that some of bits might get corrupted



- Possible solution?
  - Use an error-correcting code to encode the message with O(n) bits as long as < 1/4 bits are received incorrectly
  - Discovered by Shannon in 1948

BACKGROUND

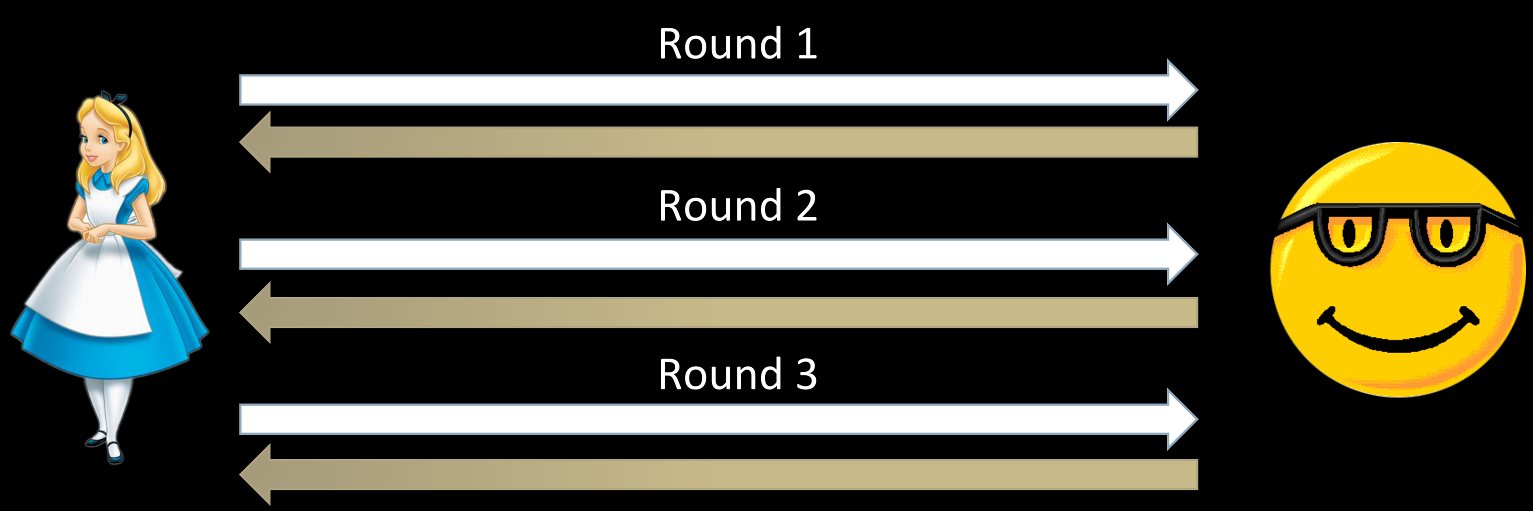
- Alice and Bob want to send each other multiple *1*-bit messages but are worried that some of bits (messages) might get corrupted



- Can no longer use Shannon's error-correcting codes
  - Must use protocol that allows for "interactive communication"

INTERACTIVE COMMUNICATION

- Communication where each party sends many messages back and forth and can change what they want to say based on the other party's responses
- Proceeds in rounds
  - Each party takes turns sending/receiving



INTERACTIVE COMMUNICATION

Schulman's *Coding for Interactive Communication* (1996)

- Any protocol with *N* bits of interactive communication can be encoded with a constant-sized alphabet
- Bits exchanged in the protocol are determined one at a time, dynamically in the course of the interaction
- Purpose was to model a noisy communication link between two interacting computer processors
- Tolerates at most error rate of 1/240

INTERACTIVE COMMUNICATION

Braverman and Rao's *Towards coding for maximum errors in interactive communication* (2011)

- New protocol that builds and improves on Schulman's work and also uses a constant-sized alphabet of size $O(1/\varepsilon)$
- Tolerates at most error rate of $1/4 - \varepsilon$
- Error rate is based off "worst-case scenario" errors, e.g., can have an adversary that knows the protocol choose which bits to corrupt

COMPARISON

Schulman's error rate: 1/240

j o y a q c d x l p a t v j r u k q z q y n d g e n j c q g a c t x k w z
o j p e b w e z m k l n j k n i p b e a x i z t w y q b i x c t r n p e z
w s k e v j s i u g l r l u c m i g m i t p l d m j n u i u a y g b i u w s
z q u l d h z t n y i r v c t f j v i k o d u p s q r h b v e e d l m d d k
a u b v v a p r t a m f s k y o d v k y p n p c a r a u p n h e u r x y
k o h s g h s s p z w k z f t q w e m r r q c e m l t v d h q q r o k d
s h c u m t a t e q a t o g w s q b e h m x j f v m c d k u e k j b n
k l s j x w a r s n z p l m m h z n b r z x f a x p x s o g b q k s i u e
p p o l l w z l h j t j p x b i n y v i z w s s q r e i e c w u y s j w g i f
y x v r h n j x r r f m p p q l i w v f f a a a i u o f b d q u d n d y u
u s p g k x u c j a u g u n v e v l b m k e p i e b e i b o h r i r v q x
p w c i k w x f b n j f n x v n e q v r u e j s a c v k a x p y i f o j t f
x n x l k x m q z n u w z h x l t t q h q s v i t z v f j z z t b o p

COMPARISON

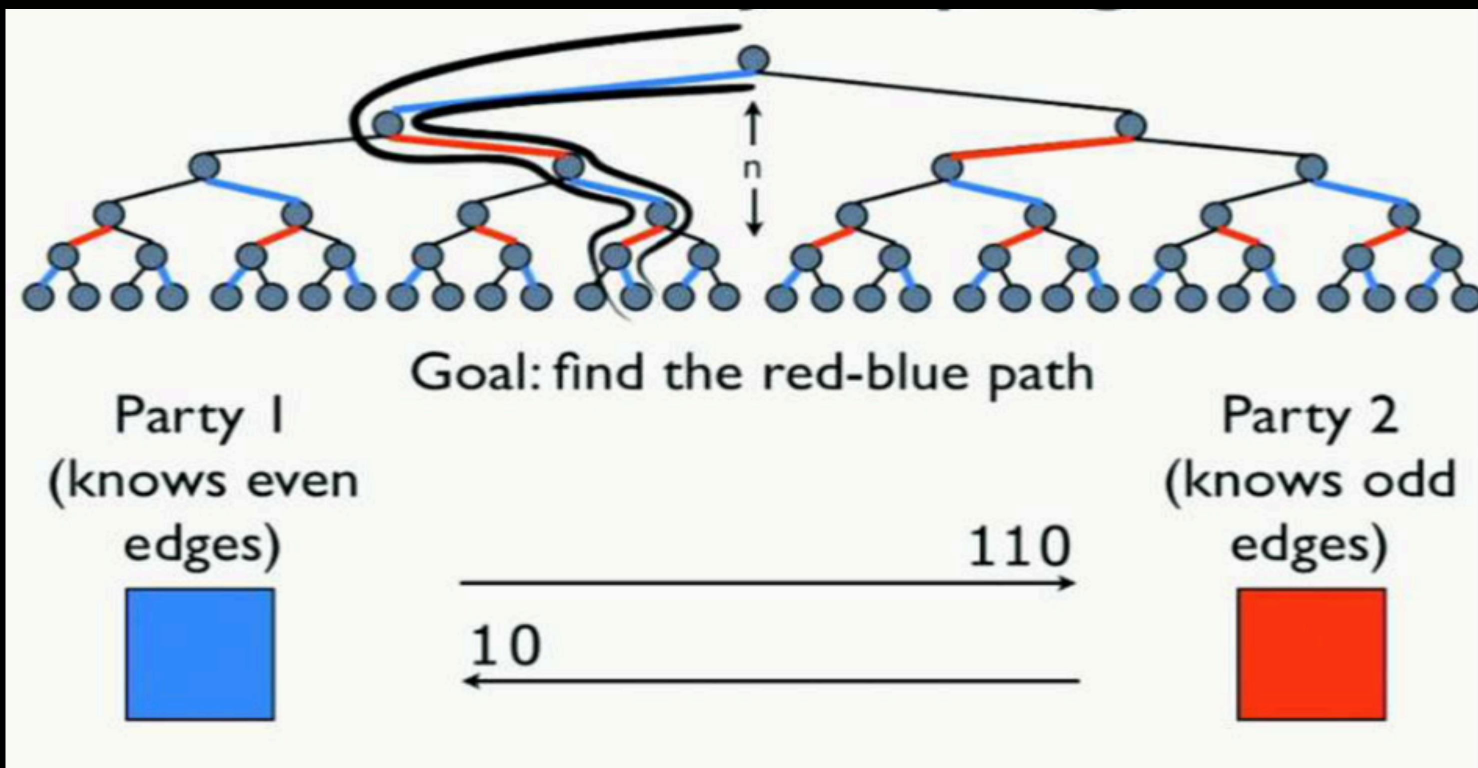
Braverman and Rao's error rate: 1/4 - ε

j o y a q c d x l p a t v j r u k q z q y n d g e n j c q g a c t x k w z
o j p e b w e z m k l n j k n i p b e a x i z t w y q b i x c t r n p e z
w s k e v j s i u g l r l u c m i g m i t p l d m j n u i u a y g b i u w s
z q u l d h z t n y i r v c t f j v i k o d u p s q r h b v e e d l m d d k
a u b v v a p r t a m f s k y o d v k y p n p c a r a u p n h e u r x y
k o h s g h s s p z w k z f t q w e m r r q c e m l t v d h q q r o k d
s h c u m t a t e q a t o g w s q b e h m x j f v m c d k u e k j b n
k l s j x w a r s n z p l m m h z n b r z x f a x p x s o g b q k s i u e
p p o l l w z l h j t j p x b i n y v i z w s s q r e i e c w u y s j w g i f
y x v r h n j x r r f m p p q l i w v f f a a a i u o f b d q u d n d y u
u s p g k x u c j a u g u n v e v l b m k e p i e b e i b o h r i r v q x
p w c i k w x f b n j f n x v n e q v r u e j s a c v k a x p y i f o j t f
x n x l k x m q z n u w z h x l t t q h q s v i t z v f j z z t b o p

DEFINITIONS

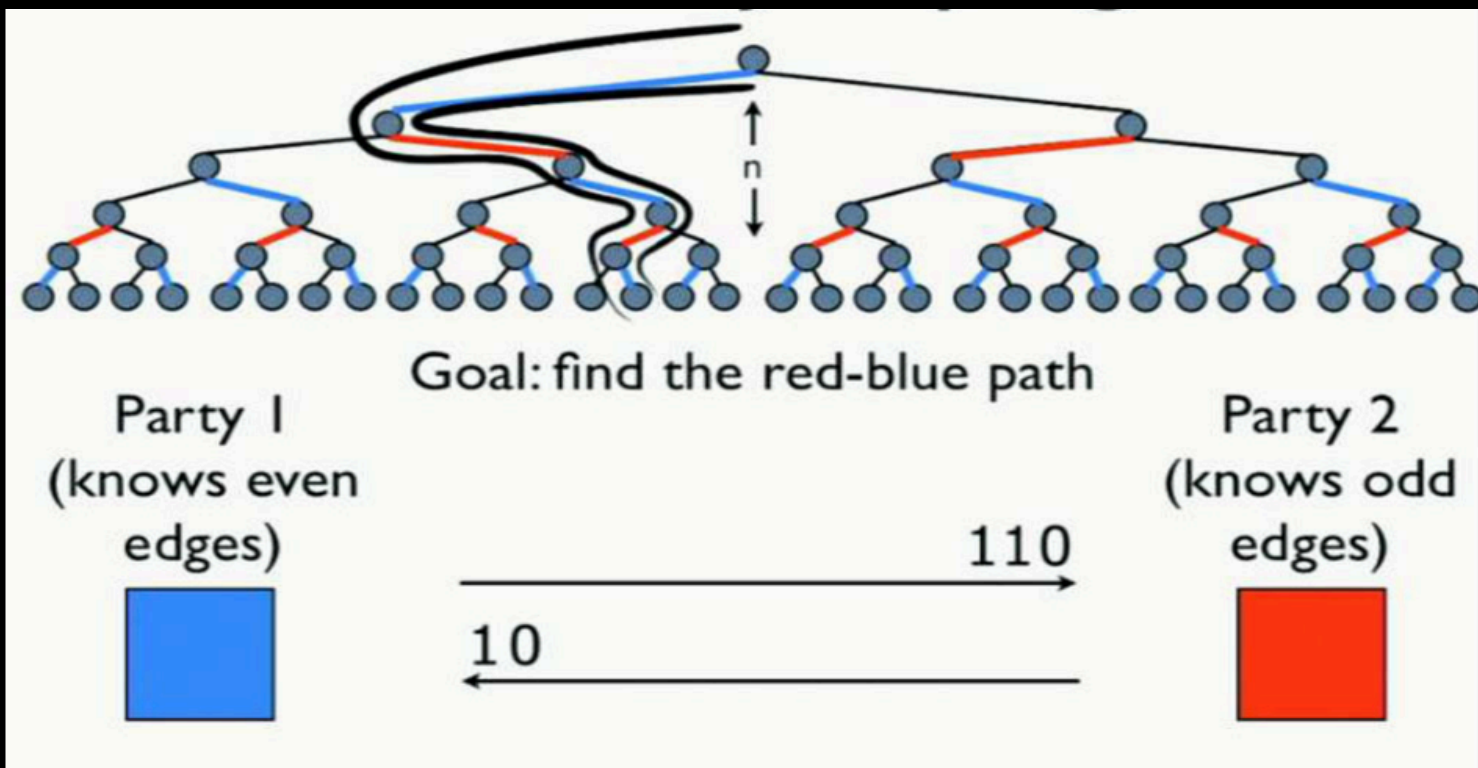
POINTER JUMPING

- Binary tree of height *n*
- Goal: find the red-blue path
  - Represents communication between the two parties



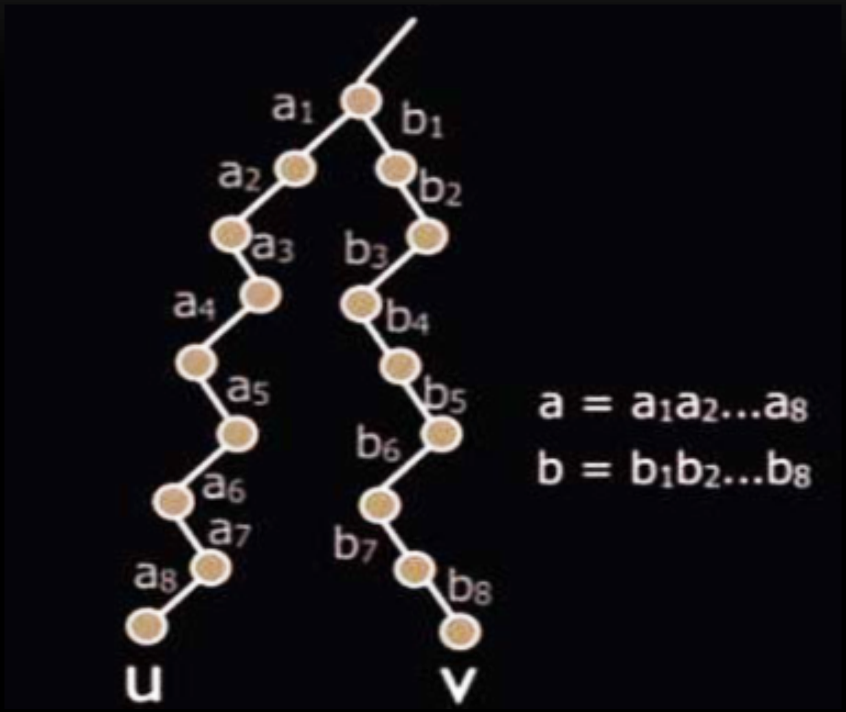
POINTER JUMPING

- 0 = left edge; 1 = right edge
- Party 1 communicates the bits 0, 1, 1
- Party 2 communicates the bits 0, 1



d-ARY TREE CODES

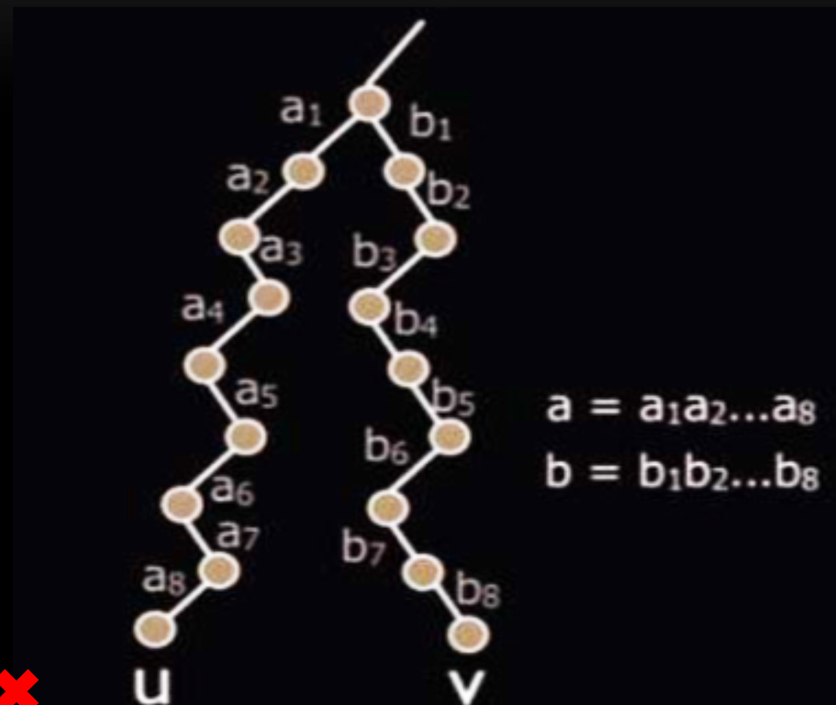
- *d*-ary tree – each vertex has *d* children
- Each edge labeled by a symbol from our alphabet Σ
- $L(u, v)$ = distance from u and v to their closest common ancestor
- $\Delta(u, v)$ = hamming distance between symbols in paths from root to u and v
- We say tree code has “distance” $(1 - \varepsilon)$



Tree code restriction:
$\Delta(u, v) < (1 - \varepsilon)L(u, v)$

d-ARY TREE CODES

- Example:
  - Let $\varepsilon = 1/6$
  - a and b must differ in more than $(1 - \varepsilon)*8 = (5/6)*8 = 6.667$ places
  - Bad coding:
    - a = 10000011
      b = 11110100 } $\Delta(a, b) = 6$ ✖
  - Good coding:
    - a = 10001011
      b = 11110100 } $\Delta(a, b) = 7$ ✔

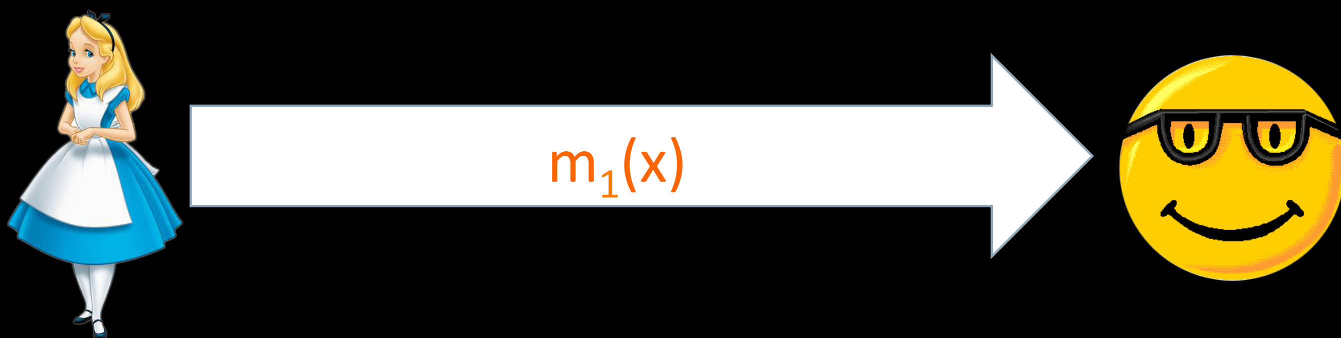


Tree code restriction:
$\Delta(u, v) < (1 - \varepsilon)L(u, v)$

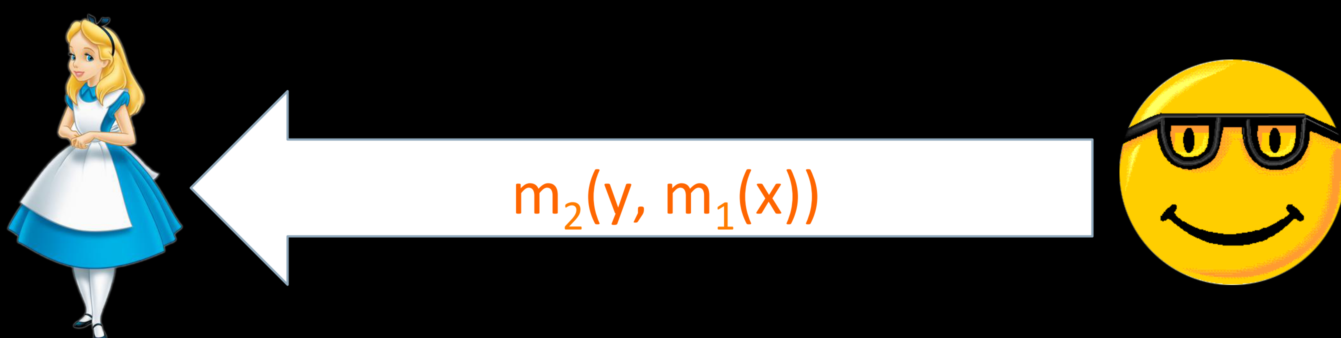
PROTOCOL

PROTOCOL

- Alice wants to send message x to Bob
  - She sends it encoded as $m_1(x)$



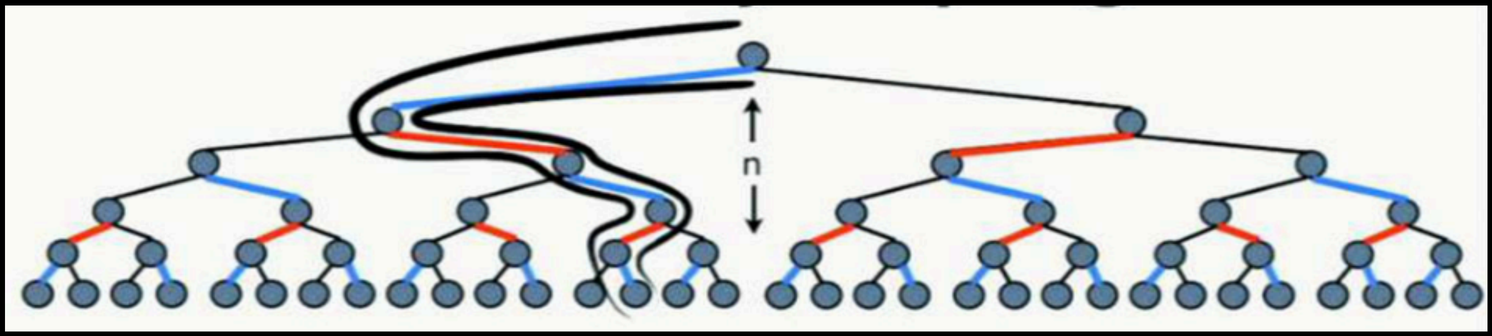
- Then Bob wants to reply with message y
  - He sends it encoded as $m_2(y, m_1(x))$



PROTOCOL

- Takes a given binary protocol (modeled by a binary tree T) between Alice and Bob and converts it into one that is error resistant, allowing an error rate of $1/4 - \varepsilon$
- Alice knows set of edges X in T of even depth
- Bob knows set of edges Y in T of odd depth
- Goal is to communicate the vertex reachable by the edges in X and Y

Binary tree T



PROTOCOL

- Construct a d-ary tree code and encoding/decoding function that converts between our tree code alphabet, Π, and our protocol's alphabet, Σ.
- Alice: knows set $X = \cup X_i$, receives set Y'_i from Bob each round
- Bob: knows set $Y = \cup Y_i$, receives set X'_i from Alice each round
- For $i = 1,\dots,H/\varepsilon$, where H is the height of our binary tree T:
  - Alice calculates her best interpretation of X and Y'_i, and transmits an encoding of this interpretation, $m(X \cup Y'_i)$
  - Bob calculates his best interpretation of Y and X'_i, and transmits an encoding of this interpretation, $m(Y \cup X'_i)$
- At the end, both Alice and Bob will have calculated the vertex reachable by $X \cup Y$

ANALYSIS

ANALYSIS

Schulman showed:

- For every $0 < \varepsilon < 1$,
  a d-ary tree code with distance $(1 - \varepsilon)$ exists
  using an alphabet of size $d^{O(1/\varepsilon)}$
- An explicit construction of tree codes, but takes $\Theta(1/\log n)$ time

ANALYSIS

Braverman and Rao showed:

- For every $0 < \varepsilon < 1$, any binary protocol can be simulated using a protocol with an alphabet of size $O(1/\varepsilon)$ if the error rate is at most $1/4 - \varepsilon$
- For every $0 < \varepsilon < 1$, any binary protocol can be simulated using a binary protocol if the error rate is at most $1/8 - \varepsilon$

But still haven't found a constant-rate construction of tree codes to date – remains a major open problem.

REFERENCES

- C. Shannon. A mathematical theory of communication. Bell Systems Technical Journal, pages 623-656, 1948.
- L. Schulman. Coding for interactive communication. IEEE Transactions on Information Theory, pages 17451756, 1996.
- M. Braverman and A. Rao. Towards coding for maximum errors in interactive communica- tion. STOC, 2011.