Interactive Coding - Lecture 1

Challenge: Can you preserve an interaction when channel is (adversarially/randomly) noisy?

Example: Two players playing online chess over noisy channel.

Interaction:

- Two players A and B.
- Alice has a collection of functions $\Pi_A = \left\{\Pi_A^{(i)}\right\}$. Similarly, Bob has Π_B.
- $\Pi_A^{(i)} : \left(\{0,1\}^*\right)^{i-1} \to \{0,1\}^* \cup \{\perp\}$ for odd i.
- $\Pi_B^{(i)} : \left(\{0,1\}^*\right)^{i-1} \to \{0,1\}^* \cup \{\perp\}$ for even i.
- $\Pi_A^{(i)}(w_1, \dots, w_{i-1})$ specifies what Alice would say in round i after history of transcript $w_1, \dots, w_{i-1}$.
- $\Pi_A^{(k)}(w_1, \dots, w_{i-1}) = \perp$ means end of interaction. Output of the interaction is the entire transcript $w_1, \dots, w_k$.
- We'll consider deterministic protocols, so w_i are deterministic functions of $w_1, \dots, w_{i-1}$.
- In general $w_i \in \{0,1\}^*$, but we will consider $w_i \in \{0,1\}$, by stretching interaction by a factor of 2.
- In general, length could be variable. But we will consider fixed length k.

Noisy interactive coding:

- w_i is received as w_i'. For α fraction of the communication, i.e. αn total errors (can consider adversarial or random errors).
- Without correction: Immediately changes all future messages & so entire interaction can change (recall: chess example).
- Attempt 1: Standard Error correction in every round. Adversary can change $E(w_i)$ to $E(w_i')$ and get same effect. Can work in random error model with $O(\log n)$ blow up in communication.
- Need better solution!

Solution Concept: Interactive Coding with α-fraction errors.

- $(\Pi_A, \Pi_B) \mapsto ((\sigma_A, f_A), (\sigma_B, f_B))$
- For every sequence of $a_1, a_2, \dots, a_n$ and $b_1, \dots, b_n$ s.t.
  - $a_i = \sigma_A^{(i)}(a_1, \dots, a_{i-1})$ for odd i.
  - $b_i = \sigma_B^{(i)}(b_1, \dots, b_{i-1})$ for even i.
  - $\#\{i : a_i \neq b_i\} \leq \alpha n$.

  it holds that $f_A(a_1, \dots, a_n) = f_B(b_1, \dots, b_n) = w_1, \dots, w_k = \text{Output}(\Pi_A, \Pi_B)$.
  Here $(a_1, \dots, a_n)$ is Alice's version of the transcript; $(b_1, \dots, b_n)$ is Bob's version.
- Note that σ_A and σ_B are possibly acting on different strings!

Tree Codes

Defn: $T : [d]^n \to [q]^n$ is a (d, q, δ)-tree code if

- $T(m_1, \ldots, m_n)_i$ depends only on $m_1, \ldots, m_i$.
  Thus, another way to interpret T is using label $L : [d]^{\leq n} \to [q]$,
  and $T(m_1, \ldots, m_n) = L(m_1) \circ L(m_1, m_2) \circ \cdots \circ L(m_1, \ldots, m_{n-1})$.
  (Figure: Labelling arcs of a d-ary tree.)

- For any $m_1, \ldots, m_n$ and $m'_1, \ldots, m'_n$ such that $m_1 = m'_1, \ldots, m_i = m'_i$ and $m_{i+1} \neq m'_{i+1}$, it holds,

  $$\Delta(T(m_1, \ldots, m_n), T(m'_1, \ldots, m'_n)) \geq \delta(n - i)$$

  Note that prefix necessarily agrees.

- *Remark:* This is unlike regular coding theory where $[q]^k \to [q]^n$. We want n coordinates of input as well. We compensate by making output alphabet larger.

- Allows, decoding as long all suffixes have small fraction of errors. If $(s_1, \ldots, s_i) = T(m_1, \ldots, m_i)$, suppose $r_1, \ldots, r_i$ is such that $\Delta((s_{j+1}, \ldots, s_i), (r_{j+1}, \ldots, r_i)) \geq \delta(i - j)/2$ for all j, then $D(r_1, \ldots, r_i) = (m_1, \ldots, m_i)$.

  Alternately, suppose $(s_1, \ldots, s_i) = T(m_1, \ldots, m_i)$, suppose $r_1, \ldots, r_i$ decodes to $m'_1, \ldots, m'_i$ where $m_1 = m'_1, \ldots, m_j = m'_j$, but $m_{j+1} \neq m'_{j+1}$. Then, $\Delta((s_{j+1}, \ldots, s_i), (r_{j+1}, \ldots, r_i)) \geq \delta(i - j)/2$.

Tree codes exist!

- Random "tree" functions fail with high probability (close to 1, in fact).

- Random linear code works!

  $$T(m) = \begin{bmatrix} m_1 & \cdots & m_n \end{bmatrix} \begin{bmatrix} a_1 & a_2 & \cdots & a_n \\ & a_1 & \cdots & a_{n-1} \\ & & \ddots & \vdots \\ & & & a_1 \end{bmatrix}$$

  we interpret $a_i \in \mathbb{F}_q$ and $m_i \in [d] \subseteq \mathbb{F}_q$. That is,
  $T(m)_1 = a_1 m_1$
  $T(m)_2 = a_2 m_1 + a_1 m_2$
  $\cdots$
  $T(m)_i = a_i m_1 + a_{i-1} m_2 + \cdots + a_1 m_i$.

- Proof sketch: For any $m_1, \ldots, m_j$ and $m'_1, \ldots, m'_j$, such that $m_1 \neq m'_1$, the event of $T(\mathbf{m})_i \neq T(\mathbf{m}')_i$ happens with probability $1 - 1/q$ and is independent for different i.

  Only depends on $(m_1 - m'_1), \ldots, (m_j - m'_j)$. Union bound over different d^j different path differences of length j. Automatically handles all pairs of paths, which diverge in the last j positions.

Using Tree Codes

Two approaches:

- Schulman : "Local" approach. More natural, but weaker analysis.
- Braverman-Rao : "Holistic" approach. Less natural, but less wasteful (provably).

Common features:

- Alice and Bob maintain states $S_A^{(i)}$ and $S_B^{(i)}$ for $i = 1, \cdots, N$ for some $N = O(n)$.
- Sequence of states $S_A^{(1)}, \ldots, S_A^{(t)}$ compressed into $x^{(1)}, \ldots, x^{(t)}$ in a prefix respecting way.
- On moving to state $S_A^{(t+1)}$, communicate $L(x^{(1)}, \ldots, x^{(t+1)})$ to Bob.

Differences:

- Description of state?
- What kinds of transitions are possible?
- Rules for the transitions?
- Analysis? How many fraction of errors tolerated?

Pre-processing for Schulman's protocol:

- Alice and Bob exchange only 1 bit in each round simultaneously. (can be done with another factor 2 blow up). This makes the situation symmetric w.r.t. Alice and Bob.
- Protocol communicates fixed n bits in total (where n is known to Alice and Bob). They extend the protocol up to $O(n)$ rounds by transmitting 0's after the end.

Schulman's protocol preliminaries:

- Original protocol is a 4-ary tree, where in each round Alice and Bob exchange 1 bit each.
- $S_A^{(i)}$ is the node reached in Π, after i rounds.
- Evolution will be such that $S_A^{(i)} \in S_A^{(i-1)} + \{00, 01, 10, 11, H, B\}$.
- $x_A^{(i)}$ is the transition made in going from $S_A^{(i-1)}$ to $S_A^{(i)}$, in addition to the next bit to be sent by Alice.
- Communicate $L(x_A^{(1)}, \ldots, x_A^{(i)})$ to Bob.
  Note that $d = 12$, since $x_A^{(i)} \in \{00, 01, 10, 11, H, B\} \times \{0, 1\}$.

Actual protocol:

- Initial state $S_A^{(1)}$ is at root. $x_A^{(1)} = (H, a_1)$.
- Repeat $N = O(n)$ times. In iteration i:
  - Transmit $L(x_A^{(1)}, \ldots, x_A^{(i)})$ to Bob.

  - Given received sequence from Bob, obtain $\overline{y}_B^{(1)}, \ldots, \overline{y}_B^{(i)}$ (this is Alice's guess for $y_B^{(1)}, \ldots, y_B^{(i)}$).
  - Compute $\overline{S}_B^{(i)}$ and the next bit b_i that Bob sent.
  - Depending on relation between $S_A^{(i)}$ and $S_B^{(i)}$, do
    * If $S_A^{(i)} = \overline{S}_B^{(i)}$, then move $S_A^{(i)}$ to child given by (a_i, b_i). In this case $x_A^{(i+1)} = ((a_i, b_i), a_{i+1})$.
    * If $S_A^{(i)}$ is ancestor of $\overline{S}_B^{(i)}$, then hold. In this case, $x_A^{(i+1)} = (H, a_i)$.
    * If $\overline{S}_B^{(i)}$ is ancestor of $S_A^{(i)}$, then back up one step. In this case $x_A^{(i+1)} = (B, a')$, where a' is the bit sent by Alice at the parent of $S_A^{(i)}$.

Analysis:

- Let the true states of Alice and Bob be S_A and S_B at time i. Let S be the least common ancestor of S_A or S_B.
- Define potential $\Phi(i) = \text{depth}(S) - \max\{\text{depth}(S_A) - \text{depth}(S), \text{depth}(S_B) - \text{depth}(S)\}$. This is depth of S minus the distance from S to the further of S_A and S_B.
- Define good round as one where both Alice and Bob decode the entire history of x_A and y_B correctly.
- In good round, potential increases by 1. In bad round, potential decreases by at most 3.
- If N_g (resp. N_b) is number of good rounds (resp. bad rounds).
- Then $\Phi(N) \geq N_g - 3N_b = N - 4N_b$.
- **Key Lemma** (about tree codes): Let T be a tree code of distance 0.7 (i.e. $\geq 2/3$). Suppose $(s_1, \ldots, s_n) = T(m_1, \ldots, m_n)$. Let $(r_1, \ldots, r_n)$ be such that $\Delta(\mathbf{s}, \mathbf{r}) = \beta n$. Let I be the set of coordinates such that $D(r_1, \ldots, r_i) \neq (m_1, \ldots, m_i)$. Then, $|I| \leq 3\beta n$.

  Proof. If an error happens on coordinate i, include i in I. Additionally, include 2 more coordinates after that in I as *potentially bad*. If there are errors on the coordinates that were intended to be included in I, then include coordinates after that. Every coordinate not in I has the property that every suffix has at most $1/3$ fraction of errors. Hence, every unmarked node is decoded correctly. Hence $|I| \leq 3\beta n$.

  Remark: If we choose a tree code of distance $1 - \varepsilon$, then we can generalize to saying that $|I| \leq (2\beta/(1-\varepsilon)) \cdot n$.
- Finally, finishing the proof. Say $\beta_A N$ of Alice's messages are corrupted, and $\beta_B N$ of Bob's messages are corrupted. Note, that overall error fraction is $\beta = (\beta_A + \beta_B)/2$. From lemma, there are at most $(3\beta_A)N$ rounds where Bob decodes incorrectly; $(3\beta_B)N$ rounds where Alice decodes incorrectly. So, at most $(3(\beta_A + \beta_B))N = (6\beta)N$ rounds in which at least one party decodes incorrectly.
- Thus, $N_b \leq 6\beta N$. Thus, potential Φ at the end is at least $N(1 - 24\beta)$.
- Suppose $\beta = 1/48$. Then, potential Φ at the end is at least $N/2$. That is, choose $N > 2n$.

- Suppose $\beta = 1/24 - \varepsilon$, then potential is at least $24\varepsilon N$. That is, choose $N > n/24\varepsilon$.
- Can be further improved to $1/16 - \varepsilon'$ by using tree codes with distance $1 - \varepsilon$.
  (Needs to be checked: Schulman showed an error correction of 1/240.)

Summary of Schulman's solution:

- Corrects $\Omega(1)$ fraction errors.
- Not maximal fraction?
- Tree codes exist. But constructive? Decoding is brute force.
- Weakness: Progress is made only when entire transcript is decoded correctly. Moreover, 3x negative progress is made otherwise. Can we avoid the negative progress?

Current state of the art:

- Exact capacity (even with random errors) unknown.
- Maximal fraction of errors? Essentially known [Braverman-Rao].
- Maximal error fraction over binary alphabet?
- Known if adversary has separate budget for Alice and Bob corruptions.
- Rate as error goes to 0. Essentially known. Rate $\approx 1 - \widetilde{O}(\sqrt{\varepsilon})$. [Kol-Raz], [Haeupler].
  In contrast to one-way communication where rate is $1 - \widetilde{O}(\varepsilon)$.
- Polynomial time encoding + decoding: essentially known [Brakerski-Kalai], while losing out on errors tolerated.

Interactive Coding - Lecture 2