Communication Amid Uncertainty

Madhu Sudan
Microsoft Research

Based on joint works with Brendan Juba, Oded Goldreich, Adam Kalai, Sanjeev Khanna, Elad Haramaty, Jacob Leshno, Clement Canonne, Venkatesan Guruswami, Badih Ghazi, Pritish Kamath, Ilan Komargodski and Pravesh Kothari.

Context in Communication

- Sender + Receiver share (huuuge) context
  - In human comm: Language, news, Social
  - In computer comm: Protocols, Codes, Distributions
  - Helps compress communication

- Perfectly shared $\Rightarrow$ Can be abstracted away.
- Imperfectly shared $\Rightarrow$ What is the cost?
  - How to study?

Communication Complexity

The model (with shared randomness)

x

$f:(x,y) \mapsto \Sigma$

$R = \$\$\$$

y

Alic

Usually studied for lower bounds.

This talk: CC as +ve model.

Bob

$CC(f) =$ # bits exchanged
by best protocol

$f(x,y)$
w.p. 2/3

Modelling Shared Context + Imperfection

1

- Many possibilities. Ongoing effort.
- Alice+Bob may have estimates of x and y.
  - More generally: x, y correlated.
- Knowledge of f – function Bob wants to compute
  - may not be exactly known to Alice!

3

- Shared randomness
  - Alice + Bob may not have identical copies.

2

Part 1: Uncertain Compression

Specific Motivation: Dictionary

$$
\begin{aligned}
M_1 &= w_{11}, w_{12}, \ldots \\
M_2 &= w_{21}, w_{22}, \ldots \\
M_3 &= w_{31}, w_{32}, \ldots \\
M_4 &= w_{41}, w_{42}, \ldots \\
&\ldots
\end{aligned}
$$

- Dictionary: maps words to meaning
  - Multiple words with same meaning
  - Multiple meanings to same word
- How to decide what word to use (encoding)?
- How to decide what a word means (decoding)?
  - Common answer: <u>Context</u>
- Really Dictionary specifies:
  - Encoding: context × meaning → word
  - Decoding: context × word → meaning
- Context implicit; encoding/decoding works even if context used not identical!

Context?

- In general complex notion …
  - What does sender know/believe
  - What does receiver know/believe
  - Modifies as conversation progresses.

- Our abstraction:
  - Context = Probability distribution on potential "meanings".
  - Certainly part of what the context provides; and sufficient abstraction to highlight the problem.

The (Uncertain Compression) problem

- Wish to design encoding/decoding schemes (E/D) to be used as follows:
  - Sender has distribution P on $M = \{1,2,\ldots,N\}$
  - Receiver has distribution Q on $M = \{1,2,\ldots,N\}$
  - Sender gets $X \in M$
  - Sends $E(P,X)$ to receiver.
  - Receiver receives $Y = E(P,X)$
  - Decodes to $\hat{X} = D(Q,Y)$

  - Want: $X = \hat{X}$ (provided P, Q close),
    - While minimizing $Exp_{X \leftarrow P} |E(P,X)|$

Closeness of distributions:

- P is Δ-close to Q if for all $X \in M$,

$$\frac{1}{2^{\Delta}} \leq \frac{P(X)}{Q(X)} \leq 2^{\Delta}$$

- P Δ-close to Q $\Rightarrow$ $D(P||Q), D(Q||P) \leq \Delta$.

Dictionary = Shared Randomness?

- Modelling the dictionary: What should it be?

- Simplifying assumption – it is shared randomness, so …

- Assume sender and receiver have some shared randomness R and X, P, Q independent of R.
  - $Y = E(P, X, R)$
  - $\hat{X} = D(Q, Y, R)$

- Want $\forall X,\ \Pr_R[\hat{X} = X] \geq 1 - \epsilon$

Solution (variant of Arith. Coding)

- Use R to define sequences
  - $R_1\,[1], R_1\,[2], R_1\,[3], \ldots$
  - $R_2\,[1], R_2\,[2], R_2\,[3], \ldots$
  - …
  - $R_N\,[1], R_N\,[2], R_N\,[3], \ldots.$
- $E_\Delta(P, x, R) = R_x[1 \ldots L]$, where L chosen s.t. $\forall z \neq x$

  Either $R_z[1 \ldots L] \neq R_x[1 \ldots L]$

  Or $P(z) < \frac{P(x)}{4^\Delta}$
- $D_\Delta(Q, y, R) = \mathrm{argmax}_{\hat{x}}\ \{Q(\hat{x})\}$ among $\hat{x} \in \{\, z \mid R_z[1 \ldots L] = y\}$

Performance

- Obviously decoding always correct.
- Easy exercise:
  - $\mathrm{Exp}_X\ [E(P,X)] = H(P) + 2\,\Delta$
- Limits:
  - No scheme can achieve $(1-\epsilon)\cdot[H(P)+\Delta]$
  - Can reduce randomness needed.

Implications

- Reflects the tension between ambiguity resolution and compression.
  - Larger the ((estimated) gap in context), larger the encoding length.
  - Entropy is still a valid measure!
- Coding scheme reflects the nature of human communication (extend messages till they feel unambiguous).
- The "shared randomness" assumption
  - A convenient starting point for discussion
  - But is dictionary independent of context?
    - This is problematic.

Deterministic Compression: Challenge

- Say Alice and Bob have rankings of N players.
  - Rankings = bijections $\pi, \sigma : [N] \rightarrow [N]$
  - $\pi(i)$ = rank of i^{th} player in Alice's ranking.
- Further suppose they know rankings are close.
  - $\forall\, i \in [N]: |\pi(i) - \sigma(i)| \leq 2.$
- Bob wants to know: Is $\pi^{-1}(1) = \sigma^{-1}(1)$
- How many bits does Alice need to send (non-interactively).
  - With shared randomness – $O(1)$
  - Deterministically?
    - With Elad Haramaty: $O(\log^* n)$

Part 2: Imperfectly Shared Randomness

Model: Imperfectly Shared Randomness

- Alice $\leftarrow r$; and Bob $\leftarrow s$ where $(r, s) =$ i.i.d. sequence of correlated pairs $(r_i, s_i)_i$; $r_i, s_i \in \{-1, +1\}; \mathbb{E}[r_i] = \mathbb{E}[s_i] = 0; \mathbb{E}[r_i s_i] = \rho \geq 0$.
- Notation:
  - $isr_\rho(f)$ = cc of f with ρ-correlated bits.
  - $cc(f)$: Perfectly Shared Randomness cc. $= isr_1(f)$
  - $priv(f)$: cc with PRIVate randomness $= isr_0(f)$
- Starting point: for Boolean functions f
  - $cc(f) \leq isr_\rho(f) \leq priv(f) \leq cc(f) + \log n$
  - What if $cc(f) \ll \log n$? E.g. $cc(f) = O(1)$

$\rho \leq \tau \Rightarrow$ $isr_\rho(f) \geq isr_\tau(f)$

Imperfectly Shared Randomness: Results

- Model first studied by [Bavarian,Gavinsky,Ito'14] ("Independently and earlier").
  - Their focus: Simultaneous Communication; general models of correlation.
  - They show $isr(\text{Equality}) = O(1)$ (among other things)

- Our Results:
  - Generally: $cc(f) \leq k \Rightarrow isr(f) \leq 2^k$
  - Converse: $\exists f$ with $cc(f) \leq k \;\&\; isr(f) \geq 2^k$

Aside: Easy CC Problems

- Equality testing:
  - $EQ(x,y) = 1 \Leftrightarrow x = y$; C
- Hamming distance:
  - $H_k(x,y) = 1 \Leftrightarrow \Delta(x,y) \leq k$;
- Small set intersection:
  - $\cap_k (x,y) = 1 \Leftrightarrow$ $\mathrm{wt}(x), \mathrm{wt}(y$
  - $CC(\cap_k) = O(k)$ [Håstad Wi
- Gap (Real) Inner Produ
  - $x, y \in \mathbb{R}^n$; $|x|_2, |y|_2 = 1$;
  - GIP

Protocol:

Fix ECC E: $\{0,1\}^n \to \{0,1\}^N$

$poly(k)$ Protoc

Use common

to hash $[n] \to$

$x = (x_1, \ldots, x_n)$

$y = (y_1, \ldots, y_n)$

$\langle x, y \rangle \triangleq \sum_i x_i y_i$

Main Insight:

If $G \leftarrow N(0,1)^n$, then

$[\langle G, x \rangle \cdot \langle G, y \rangle] = \langle x, y \rangle$

[Ghazi,Kamath,S'15]: Roughly essence of perm. inv. functions

Equality Testing (our proof)

- Key idea: Think inner products.
  - Encode $x \mapsto X = E(x); y \mapsto Y = E(y); X, Y \in \{-1, +1\}^N$
    - $x = y \Rightarrow \langle X, Y \rangle = N$
    - $x \neq y \Rightarrow \langle X, Y \rangle \leq N/2$
- Estimating inner products:
  - Building on sketching protocols …
  - Alice: Picks Gaussians $G_1, \ldots G_t \in \mathbb{R}^N$,
  - Sends $i \in [t]$ maximizing $\langle G_i, X \rangle$ to Bob.
  - Bob: Accepts iff $\langle G'_i, Y \rangle \geq 0$
  - Analysis: $O_\rho(1)$ bits suffice if $G \approx_\rho G'$

Gaussian Protocol

General One-Way Communication

- Idea: All communication $\leq$ Inner Products
- (For now: Assume one-way-cc(f) $\leq\ k$)
  - For each random string R
    - Alice's message $= i_R \in [2^k]$
    - Bob's output $= f_R(i_R)$ where $f_R: [2^k] \to \{0,1\}$
    - W.p. $\geq \frac{2}{3}$ over R, $f_R(i_R)$ is the right answer.

General One-Way Communication

- For each random string R
  - Alice's message = $i_R \in [2^k]$
  - Bob's output = $f_R(i_R)$ where $f_R: [2^k] \to \{0,1\}$
  - W.p. $\geq \frac{2}{3}$, $f_R(i_R)$ is the right answer.
- Vector representation:
  - $i_R \mapsto x_R \in \{0,1\}^{2^k}$ (unit coordinate vector)
  - $f_R \mapsto y_R \in \{0,1\}^{2^k}$ (truth table of f_R).
  - $f_R(i_R) = \langle x_R, y_R \rangle$; Acc. Prob. $\propto \langle X, Y \rangle; X = (x_R)_R; Y = (y_R)_R$
  - Gaussian protocol estimates inner products of unit vectors to within $\pm\epsilon$ with $O_\rho\left(\frac{1}{\epsilon^2}\right)$ communication.

Two-way communication

- Still decided by inner products.
- Simple lemma:
  - $\exists K_A^k, K_B^k \subseteq \mathbb{R}^{2^k}$ convex, that describe private coin k-bit comm. strategies for Alice, Bob s.t. accept prob. of $\pi_A \in K_A^k, \pi_B \in K_B^k$ equals $\langle \pi_A, \pi_B \rangle$
- Putting things together:

Theorem: $cc(f) \leq k \Rightarrow isr(f) \leq O_\rho(2^k)$

Part 3: Uncertain Functionality

Model

- Alice knows $g \approx f$; Bob wishes to compute $f(x, y)$
- Alice, Bob given g, f explicitly. (Input size ~ 2^n)
- Questions:
  - What is $\approx$?
  - Is it reasonable to expect to compute $f(x, y)$?
    - E.g., $f(x, y) = f'(x)$? Can't compute $f(x, y)$ without communicating x
- Answers:
  - Assume $x, y \sim \{0,1\}^n \times \{0,1\}^n$ uniformly.
  - $f \approx_\delta g$ if $\delta(f, g) \leq \delta$.
  - Suffices to compute $h(x, y)$ for $h \approx_\epsilon f$

Results - 1

- Thm [Ghazi,Komargodski,Kothari,S.]: $\forall \epsilon > 0, \exists \delta > 0$ s.t. If f has one-way communication k, then in the (ϵ, δ) −uncertain model it has communication complexity $O(k)$.
- Main Idea:
  - Canonical protocol for f:
    - Alice + Bob share random $x_1, \ldots x_m \in \{0,1\}^n$.
    - Alice sends $f(x_1), \ldots, f(x_m)$ to Bob.
    - Protocol used previously … but not as "canonical".
  - Canonical protocol robust when $f \approx g$.

Results - 2

- Can extend model to $(x, y) \sim \mu$ for arbitrary distribution μ
- Results:
  - If μ is product distribution ($x \perp y$) then results extend.
  - Else
    - Upper bound: Multiplicative factor of $I(x, y)$
    - Lower bound: Some blowup necessary
      - $\exists \mu$ and dist. on pairs of functions (f, g) of constant comm. complexity; but computing $g(x, y)$ in the uncertain model costs $\Omega(\sqrt{n})$ bits.
- Open: Significance of above?

Conclusions

- Context Important:
  - New layer of uncertainty.
  - New notion of scale (context LARGE)

- Many open directions+questions

Thank You!