1 Pseudorandomness via iterative simplification

References

- 2012 Gopalan, Meka, Reingold, Trevisan, Vadhan
- 2015 Gopalan, Kane, Meka

We want to study randomness as a resource. Randomness is useful for cryptography, e-commerce, and algorithms.

Where do random bits come from? There are companies which sell randomness.

We should ask whether randomness is necessary for computation. A central challenge is $P\overset{?}{=}BPP$. There is strong evidence that randomness is not needed (hardness vs. randomness). If hard functions exist then P=BPP. But this doesn't prove it.

"Randomness is in the eye (computational power) of the beholder."

A pseudorandom generator stretches bits to fool a class of test functions $\mathcal{F}$. $G : B^r \to B^n$,

$$|\mathbb{P}_{x\sim B^n}[f(x)=1]-\mathbb{P}_{y\sim B^r}[f(G(y))=1]|<\varepsilon.$$

We say G ε-fools $\mathcal{F}$.

They're used in complexity, random algorithms, approximations.

We focus on bounded depth circuits and bounded space computation.

1.1 PRGs for bounded depth circuits

- Parity problem, small-error
- PRG for read-once CNF's.

AC0 (bounded depth, unlimited fan-in): There sxists a non-explicity generator with seed length $O(\ln\left(\frac{n}{\varepsilon}\right))$.

For polynomially small error the best was $O((\ln n)^2)$ even for read-once CNF's and DNF's. Why do we care about small-error? To derandomize, we only need error $\frac{1}{3}$.

Not being able to handle small error is symptomatic of bottleneck for larger depth. If there is a PRG for dept 2 with seed $O(\ln\left(\frac{n}{\varepsilon}\right))$, then NP does not have depth 3 size $2^{o(n)}$ circuits. The best by Hastad is $2^{\sqrt{n}}$ lower bound for parity. There is a matching upper bound; we need new functions and techniques.

Also lower bound for linear size circuits of log depth (trade size for depth).

GMRTV12 gives PRG for read-once CNFs with seed-length $\widetilde{O}(\ln\left(\frac{n}{\varepsilon}\right))$. (Build on ideas of CRSW10, KMN10.) In a read-once CNF, each variable appears at most once.

We sketch the construction of PRGs for CNFs.

We use the hammer of random restrictions and switching lemmas. Randomly fix $\approx 1-\frac{1}{\ln n}$ fraction of variables. Then a formula on $\frac{n}{\ln n}$ variables collapses to constant size.

Ajtai and Wigderson (1985) used pseudorandom restrictions. Problem: there is no good pseudorandom switching lemma—we don't have a good family of psuedorandom functions.

Look at mild pseudorandomness restrictions: restrict only half the bits pseudorandomly. the average function is fooled by small bias spaces.

This suggests the following iterative generator: Choose and set a fraction of variables. Repeat $O(\ln\ln n)$ times. $O((\ln n)(\ln\ln n))$ bits are used.

Restricting half the bits in a read-once CNF in n variables gives a read-once CNF in $\sqrt{n}$ variables.

The switching lemma is very subtle; it's hard to derandomize. The above claim (simple counting) can derandomize with limited independence.

Open problems.

1. Power of mild pseudorandom restrictions: what else can the generator fool? E.g., combining small-bias spaces is very powerful, e.g., for $\mathbb{F}_2$-polynomials.

2. Does the construction fool read-twice CNF's?

1.2 PRGs for small space machines

- Halfspaces, modular sums, Chernoff bounds
- One generator to fool tem all: Fourier shapes
- Outline of construction

They are both based on iterative simplification. Gradually simplify a function to get easier proofs.

We ask the space version of P vs. BPP, does RL$\overset{?}{=}$L? Nisan 92 and INW94 give a PRG with seed $O(\ln\left(\frac{n}{\varepsilon}\right)^2)$ for logspace. There are not many cases where we can beat this bound.

1. BPL in $L^{\frac{3}{2}}$ (Saks-Zhou)

2. Undirected s-t connectivity. (Reingold)

3. Regular branching programs (BRRY, BV)

GKM15 give PRG's for halfspaces, modular sums, Chernoff bounds, and most linear tests.

The two papers show new constructions that can get around previous barriers.

Halfspaces: Given $f : B^n \to B$, $f(x) = \text{sign}(\langle w, x\rangle - \theta)$.

Halfspaces have applications to perceptrons, boosting, SVM's, streaming algorithms, rounding algorithms...

We construct a PRG for halfspaces with seedlength $\widetilde{O}(\ln\left(\frac{n}{\varepsilon}\right))$. It works over $\mathbb{R}^n$ as well.

Modular sums $f : B^n \to B^n$. $f(x)$ is whether $\langle w, x\rangle \pmod m \in A$. It's a basis test function and generalizes small-bias spaces. The best previous result is $\widetilde{O}(\ln\left(\frac{n}{\varepsilon}\right) + (\ln m)^2)$. For polynomial-sized modulus, best previous is $(\ln n)^2$.

Derandomizing Chernoff bounds:

$$\mathbb{P}\left[\left|\sum X_i - \sum_i \mathbb{E}[X_i]\right|\right] \le 2\exp\left(\frac{-t^2}{4n}\right).$$

In many cases we can't affort to have complete independence. In many applications we use $O(\ln n)$-wise independence. But this nees $O((\ln n)^2)$ random bits: there is no improvement over Nisan. GKM15: we can satisfy all such tail bounds using $\widetilde{O}(\ln \binom{n}{\varepsilon})$ randomness.

One generator to fool them all: Fourier shapes. This generalizes halfspaces, modular sums, chernoff bounds, combinatorial rectangles/shapes...

Definition 1.1: A (m, n) **Fourier shape** is $f : [m]^n \to \mathbb{C}_1$ (unit complex disc) where

$$f(x) = f_1(x_1)f_2(x_2)\cdots f_n(x_n),$$

each a function to $\mathbb{C}_1$.

Why are we using complex numbers? Roots of unity captures modular sums. The discrete Fourier transform captures linear functions.

Why large domains? It's more general, and more robust: it allows nice compositions.

- ε-fool $(2, n)$ Fourier shapes $\implies$ $n^2\varepsilon$-fool halfspaces
- ε-fool $(2, n)$ Fourier shapes $\implies$ $n\varepsilon$-fool modular sums

PRG for Fourier shapes

1. A dichotomy of Fourier shapes: high-variance vs. low-variance
2. PRG for high variance Fourier shapes (generalize Naor-Naor construction)
3. Low-variance: alphabet/dimension reduction.

To build a ε-PRG for (m, n)-Fourier shapes using extra $\widetilde{O}(\ln m)$ bits, we start from a $\frac{\varepsilon}{100}$-PRG for (n, n)-Fourier shapes.

Take the square root each time, using $O(\ln m)$ bits each time.

We need to reduce the alphabet size m. Subsample each column ($\sqrt{m}$) using limited independence. Formally, generate $[m]^{\sqrt{m}\times n}$ whose columns are k-wise independent.

Just working with the definition, we want

$$\prod_{j=1}^{n}\left(\frac{1}{\sqrt{m}}\sum_{i=1}^{\sqrt{m}} f_j(X_{ij})\right) \approx \prod_{j=1}^{m}\mathbb{E}[f_j].$$

Denote the estimated column average by Y_j. We want $\mathbb{E}\left[\prod_{j=1}^{n}\right] \approx \prod_{j=1}^{n}\mathbb{E}[Y_j]$.

Lemma 1.2: Let $Y_i \in \mathbb{C}_1$ be k-wise independent. Then

$$\left|\mathbb{E}\left[\prod_{j=1}^{n} Y_j\right] \approx \prod_{j=1}^{n}\mathbb{E}[Y_j]\right| \precsim \left(\sum_{i=1}^{n}\sigma^2(Y_i)\right)^{0.1k}.$$

The error is very small if the variance is small. Subsampling reduces variance drastically.

Dimension reduction

What about $(2, n)$? We want a ε-PRG for (n, n)-Fourier shapes. Given $O(\ln\left(\frac{n}{\varepsilon}\right))$ random bits, it suffices to get a $\left(\frac{\varepsilon}{100}\right)$-PRG for $(\left(\frac{n}{\varepsilon}\right)^{10}, \sqrt{n})$-Fourier shapes. Now apply alphabet reduction to get $\frac{\varepsilon}{100^2}$-PRG for $(\sqrt{n}, \sqrt{n})$ Fourier shapes. $O(\ln\ln\left(\frac{n}{\varepsilon}\right))$ iterative simplifications. Total seed length $\widetilde{O}(\ln\left(\frac{n}{\varepsilon}\right))$.

Open problems. Generalize to log-space?

1. Generalize to matrix shapes: $f : [m]^n \to \mathbb{C}_1^{2\times 2}$. domain?
2. Does the PRG fool depth 2 circuits? (Large alphabets are like CSP's.)
3. Can we use these ideas for polynomial threshold functions? The best, by MZ10, is $O\left(\frac{(\ln n)}{\varepsilon^{O(1)}}\right)$.

2 Dynamics for the mean-field random-cluster model

In the random cluster model, we have $G = (V, E)$, and the probability measure over the subgraph $(V, A \subseteq E)$ is

$$\pi_{p,q}(A) \propto p^{|A|}(1-p)^{|E|-|A|}q^{c(A)}$$

where $c(A)$ is the number of components in (V, A). This is the unifying framework for studying several interesting distributions.

- When $q = 1$, we get the bond percolation model, For $G = K_n$, this is $G(n, p)$.
- For integer $q \geq 2$, it is dual to the ferromagnetic q-state Potts model.
- For $q \to 0$, we get weak limits UST,...

Focus: Markov chains on random-cluster configurations with stationary distribution $\pi_{p,q}$.

Chayes-Machta dynamics: Given a random cluster configuration $A \subseteq E$:

1. Activate each connected component of A independently with probability $\frac{1}{q}$.
2. Act each acive edge with probability p. (?)

It's non-local dynamics.

Now we look at the specific mean-field model, $G = K_n$. Hopefully after we understand this we can extend to larger classes of graphs.

If $p = \frac{\lambda}{n}$ then there exists $\lambda_c(q)$ such that w.h.p.,

- $\lambda < \lambda_c(q)$: all components have size $O(\ln n)$.
- $\lambda > \lambda_c(q)$: there is a component of size $\sim \theta_r n$.

The critical value is known (Bollobas et al., 1996).

Mixing time: number of steps T_{mix} until total variation distance from $\pi_{p,q}$ is small $\leq \frac{1}{4}$, starting from initial configuration.

Swendsen-Wang: for integer q. Mixing time of SW dynamics for $q = 2$ fully understood, partially understood for $q \geq 3$.

Theorem 2.1: For $q > 1$, $p = \frac{\lambda}{n}$, mixing time of CM dynamics is $\exp(\Omega(\sqrt{n}))$ for $\lambda \in (\lambda_L, \lambda_R)$, $\Theta(\ln n)$ outside.

...Second order phase transition for $1 < q \leq 2$. For $q > 2$, $\lambda = \lambda_c$, bimodal! λ_L, λ_R are where the modes disappear.

Theorem 2: In mean-field with $q > 1$.

Technique:

- couple 2 copies $(X_t), (Y_t)$ of the CM dynamics, starting from arbitrary inital configurations X_0, Y_0. If $\mathbb{P}(X_t \neq Y_t)$, then $T_{mix} \leq T$.

1. Independent evolution until both have largest component of correct size. (Drift analysis on size of largest component.)

2. Coupled evolution of (X_t), (Y_t). (Couple activation of components in a way that both active subgraphs have the same size whp. Couple edge resampling step using arbitrary bijection between active edges. So same component structure in updated part whp. Need $O(\ln n)$ consecutive successes.)

Drift function: Consider $\phi(\theta) - \theta$. Drift always has desired sign. (???)

2: First part of activation creates a discrepancy $D = o(\sqrt{n})$. Correct D using coupling of binomial distributions on I_X, I_Y (?).

- Mixing time of CM dynamics in mean-field

- Mixing time of heat-bath dynamics in $n \times n$ boxes of $\mathbb{Z}^2$.

3 Swendsen-Wang algorithm on the mean-field Potts model

Generalization of Ising model. Given spins $[q]$ and a parameter (inverse temperature) $B > 0$ (ferromagnetic for $B \geq 1$, antiferromagnetic for $B < 1$), $w(\sigma) = B^{m(\sigma)}$ where $m(\sigma)$ is the number of monochromatic edges.

We want to sample from μ.

Glauber dynamics uses local dynamics (single-site updates). Choose a random vertex; sample from distribution induced by neighbors. Slow to converge at low temperatures (exponentially many steps).

Swendsen-Wang: non-local dynamics for Potts model.

$B_c(\mathbb{Z}^2)$: uniqueness threshold:

1. if $B < B_c(\mathbb{Z}^2)$, then all monochromatic components are finite a.s.

2. if $>$, then infinite monochromatic component a.s.

"Correlation phenomenon." Thi affects the mixing time: shoots from $n \ln n$ to exponential.

Alternative dynamics potentially rapid mixing at all temperatures? Possibly Swendsen-Wang.

let M be set of monochromatic edges. Independently for $e \in M$. Percolation (delete some edges). Randomly color connected components.

Ferromagnetic Potts on Mean-field model. High/low temp is $O(\ln n)$, Intermediate temp is $\exp(n^{\Omega(1)})$.

When $\max_\alpha Z^\alpha(B)$? Ordered/disordered. Balanced B_c, B_o, B_h majority.

...

4 Towards resistance sparsifiers

Input: dense graph G.

Goal: Give sparse weighted subgraph H that approximately preserves some properties of G.

- shortest paths
- cut values
- spectra
- resistance distance

We want a metric that captures connectedness. Equivalent views

- electric voltage distance.
- random walks: commute time—expected time go to v and back to u
- Probability of appearing in random spanning tree.

Similarity measure in machine learning.

We can calculate it efficiently

$$R_G(u, v) = (\chi_u - \chi_v)^T L_G^{-1} (\chi_u - \chi_v).$$

Can we construct efficient resistance sparsifiers?

Yes, with edges $O\left(\frac{n}{\varepsilon^2}\right)$ (Batson-Spielman-Srivatava).

Can we do better? For the complete graph, spectral sparsifiers require $O\left(\frac{n}{\varepsilon^2}\right)$, but resistance with $O\left(\frac{1}{n}\right)$. On expanders, resistance metric is essentially determined by vertex degrees.

Do more graphs have efficient resistance sparsifiers? Yes for dense regular expanders

Theorem 4.1: $\Omega(n)$ regular expander has $(1+\varepsilon)$-resistance of $\widetilde{O}\left(\frac{n}{\varepsilon}\right)$. Every $\Omega(n)$ regular expander contains $\operatorname{poly}\log(n)$ regular expander as subgraph.

Decompose G into disjoint Hamiltonian cycles or random matchings. CHoose a uniformly random subset of them to form H.

Analysis uses the Cut-matching game by Khandekar-Rao-Vazirani 2006. Start with empty graph on n vertices.

1. Cut player chooses bisection

2. Matching adds perfect matching across bisection.

The cut player goal: construct expander. Matching: delay.

Cut player can win within $O((\ln n)^2)$ rounds.

Warm up: degree $> (\frac{3}{4} + \delta)n$. Find sparse regular subgraph of G.

Claim: G contains perfect matching across any bisection. Play cut-matching: Matching returns bisection given by claim. Resulting H is $O((\ln n)^2)$-regular expander subgraph of G.

Generalize to cut-weave game. Given a bisection, a weave is a graph in which every vertex has a incident edge across the bisection. Play the cut weave game. The weave player adds a r-regular weave across the bisection. Cut plaer cna win within $O(r(\ln n)^2)$ rounds.

Now assume deg $> (\frac{1}{2} + \delta)n$. Suppose G is D-regular with $D > (\frac{1}{2} + \delta)n$. G decomposes into disjoint Hamiltonian cycles.

Claim: for any bisection in G, we get a weave by choosing $O(\ln n)$ uniformly random cycles from the decomposition. Proof: set cover.

Play the cut-weave game with $r = \ln n$. Weave samples random cycles to form a weave.

Resulting H is $O((\ln n)^4)$-regular expander subgraph of G.

Extension to $D = \Omega(n)$: more technical details.

All the weave player is subsample. Oblivious to cut player. Don't simulate cut player.

Extend to more graphs? More direct analysis?

5 Approximating TSP

Find the shortest tour. Assume it's 2-connected; else we can break up the graph at a cut point.

Results:

1. Size $\frac{4n}{3}$ in a graph with a spanning tree and a simple cycle on its odd nodes.

2. Size $\frac{9n}{7}$ in cubic bipartite graphs

3. Size $(1 + O\left(\frac{1}{\sqrt{d}}\right))n$ in d-regular graphs.

Main ideas for short tours

1. Augment spanning tree with carefully carefully edges.

2. Delete carefully chosen edges from the whole graph (accounting shows it's small).

3. Augment cycle cover with few cycles.

4. Augment path covers with few paths. (Connect up using spanning tree.)

General bounds:

In every connected, the shortest path is between n and $2n-2$. Proof: spanning tree.

Christofides, 1976 gives a $\frac{3}{2}$ approximation to graph TSP (also works for metric TSP). Tour is union of minimum spanning tree and minimum T-join on odd-degree vertices. Gives connected Eulerian graph.

T-join: in induced subgraph of this edge set, collection of all odd-degree vertices is T.

Find T-join by solving matching problem. This is solvable in polynomial time.

The minimum T-join is $\le \frac{\text{opt}}{2}$. Split optimal tour into 2 pars: between T-nodes and others, minimal is $\le 2n$.

Observation: if G has spanning tree and simple cycle C on odd nodes, then tour of length $\frac{4n}{3}$.

- If $|C| > \frac{2n}{3}$ then contract cycle and double remaining spanning tree.
- If $|C| < \frac{2n}{3}$, use Christofides's idea and take shorter of even and odd segments.

Cor: If G has Hamiltonian path, then it has tour of length $\frac{4n}{3}$.

Approximating TSP: $\frac{3}{2}$ is still best approx ratio of metric TSP. Worst integrality gap example is $\frac{4}{3}$ (max degree 3). Old graph TSP bound: $\frac{7}{5}$.

Rather than start from an arbitrary spanning tree, start with one that would give a cheap T-join. Use fraction of LP to define a distribution over spanning trees, sample one at random; it is likely to have a cheap T-join.

(2) Delete edges. (Mömke, Svensson, 1.461 approximation.) Every 3-regular 2-vertex connectedgraph has a tour of length at most $\frac{4n}{3}$.

Use probability over T-joins to fix up a tree. Delete carefully chosen edges.

Assign every edge in 3-regular 2-vertex connected graph puts it in perfect matching polytope: $\chi(\partial v) = 1$ for all v. Blossom inequalities: $\chi(\partial(S)) \ge 1$ for all $|S|$ odd. (At least 1 leaving odd set.)

Every edge inside count twice. Cancel out even from odd num. Leaving is odd. 2-connectivity then implies $|\delta(S)| \ge 3$.

Caratheodory's Theorem: G with $\frac{1}{3}$ on every edge can be written as a convex combination of a polynomial number of perfect matchings.

MS11:

1. Pick a DFS tree T with back edges B. Partition edges into 2 parts: P tree edges with back edges hanging from parent.
2. Pick matching M randomly from distribution defined by $x = \frac{1}{3}$ on $E(G)$.
3. Initialize solution H to whole graph G: Delete $M \cap B, M \cap P$, double $M \cap Q$.

Eulerian connected:

- Every node has initial degree 3. One matching edge incident is deleted or doubled: degree 2 or 4.
- Connected (bottom up): if tree edge in P deleted, backedge hanging from parent connects subtree to upper part. If back edge deleted, sibling tree edge in P connects both sides.

Number of edges:

$$\mathbb{E}|H| = \frac{3n}{2} - \frac{1}{3}\frac{n}{2} - \frac{1}{3}\frac{n}{2} + \frac{1}{3}\frac{n}{2}.$$

Hence has tour of length $\le \frac{4n}{3}$.

Sebo and Vygen 2012: find nice ear decomposition, pick set of edges based on decomposition (earmuff) to form connectedsubgraph.

Extendchosen subgraph to even supergraph by adding edges from pendant ears. If many, add T-join on odd nodes; if few, use MS.

(3) Augment cycle cover with few cycles.

Find cycle cover with few cycles. Double edges between.

$\frac{3n}{2}$ tour: cycle cover with no parallel edges (union of 2 disjoint perfect matchings). Min cycle length is 4, at most $\frac{n}{4}$ cycles. add doubles spanning tree connecting cycles.

while square, replace it with gadget. Find cycle cover in square-free graph. Expand gadgets. Put 13, 24 back. Consider all cases.

Expand back to cycle cover which does not increase number of cycles.

Subcubic graphs require $\frac{4n}{3}$ edges. (3 long paths. Duplicate...)

CLS11, 12; KR13, VZ15.

Barnette's conjecture: planar bipartite cubic graphs have Hamiltonian cycle.

Idea: replace short cycles (6-cycles, etc.)

Algorithm sketch: Organic: made up or original nodes and edges.

1. while graph contains 4-cycle or organic 6-cycle, COMPRESS. Instead of compress hexagos, compress parasites: hexagons with 3-path. Will never get back a hexagon. Final amortized average cycle length ≥ 7.

2. ...

Nisheeth Vishnoi 2012: every n-vertex d-regular graph... Why would d-regular graph have cycle cover with few cycles? How can it be found?

Matrix representation of cycle covers: Given G consider its $n \times n$ adjacency matrix A. All-1 permutation is cycle cover. Permanent of adjacency matrix is number of cycle covers. Van der werden's conjecture: permanent is large.

Few permutations with linearly many cycles. (Random permutation hsa $O(\ln n)$ cycles.) Averaging argument. $O\left(\frac{n}{\sqrt{\ln d}}\right)$.

Regularity is essential: graphsof min degree d need not have short tours. Can go between 2 parts...

If all around degree d, get nothing.

Improved bounds: get down to $\frac{1}{\sqrt{d}}$.

Our approach: find spanning tree with small set T of odd degree vertices. Find small T-join of size $O(|T|) + O\left(\frac{n}{d}\right)$.

Given T' spanning T, there is T-join on T'. (Proof: drop edges connecting even sized components.)

3-net: maximal set of vertices, no two of which at distance < 3 from each other. 3-net has $\frac{n}{d+1}$-vertices. Every vertex at distance $|le2$ from 3-net. All of T plus 3-net conected by $2|T| + \frac{2n}{d+1} - 3$ edges.

Theorem: every connected d-regular graph has a spanning tree with $O\left(\frac{n}{\sqrt{d}}\right)$ odd degree vertices. Conver by spanning linear forest.

Conjecture (Magnant and Martin): path cove number of d-regular graph is $\leq \frac{n}{d+1}$.

Arboricity: cover edges by forests. Linear arboricity: cover edges by linear forests. Conjecture: $\left\lceil \frac{d+1}{2} \right\rceil$ linear forests suffice. if true, one has at least $n - O\left(\frac{n}{d}\right)$ edges.

Alon Spencer: $\leq \frac{d+\widetilde{O}(d^{\frac{2}{3}})}{2}$. Get $d^{-\frac{1}{3}}$.

Strengthen: inductive construction of path cover. Easier for directed graphs. Keep edges/arcs coming into head, going out of tail.

d even: take Euler tour... Pair up nodes. Pu in (L, R) or (R, L). Find max matching of directed edges from L to R. use dummy edges to complete to perfect matching M.

Double, $\lg n$?

... Use LLL! Fractional linear forms.

Max/min degree within small fraction: still works.

Open: use steiner cycles for grpah-TSP, Barnette, improve additive bound, ...

6 Deletion codes in the high-rate and high-error regime

In the erasure model, symbols are replaced with '?'.

In the deletion model, symbols are deleted.

We assume no errors or insertions. Receiver knows block length. Our results will be for adversarial deletions.

Why deletions?

1. Natural model for asynchronous channels.
2. Think of deletions as dropped packets.
3. There are nice combinatorial questions.

Deletions are easy! Given $m_1, \ldots, m_n$, all we have to do is give n headers. Then we're back to the model of erasures. The alphabet grows with the length of the code. We expect deletions to happen at the bit level: what can we do over constant alphabets?

Previous work.

- Lots of work on constant number of deletions. (We're interested in a constant fraction.)
- Random deletions: for deletion probability p, capacity at least $\frac{1-p}{9}$.
- Random deletions: for $p \to 0$, capacity is $1 - h(p)$.
- Adversarial deletions: explicit good binary cocdes correcting constant fraction of deletions.

Goal: understand tradeoff between redundancy (rate) and correction capability. For fixed deletion fraction, what's the best rate we can get? This is difficult even for random deletions. So we focus on coding for extremes: low/high noise.

What's possible? Greedy construction:

- High noise: correct $1-\varepsilon$ deletion with rate $\Omega(\varepsilon)$ and alphabet size $O\left(\frac{1}{\varepsilon^3}\right)$.
- Low noise: Binary codes correct ε with rate $1-2h(\varepsilon)$.

For high note, large alphabet is necessary: with $>\frac{1}{2}$, can delete all 1's.

Theorem 6.1: (High noise) Explicit code correct $1-\varepsilon$ fraction of deletions with rate ε^2, alph $\frac{1}{\varepsilon^4}$
(High rate) ε, $1-\sqrt{\varepsilon}$

We know 2 kinds of deletion codes. Good explicit codes for large alphabets (headers). Good non-explicit codes for small alphabets (brute force). Put them together!

Outer code $B_1, \ldots$, relace B_i with block $\text{Enc}(B_i)$, inner code. We hope if both are deletion resistant, so is combined.

Erasures: How to decode concatenated code? By averaging, if not too many erasures, enough will survive.

Deletions: We want to recover each block individually, but we don't know where the blocks are. How to locate the boundaries?

High deletions: $1-\varepsilon, \text{poly}(\varepsilon), \text{poly}\left(\frac{1}{\varepsilon}\right)$.

Initial code: concatenate Reed-Solomon with headers and small alphabet code against $1-\frac{\varepsilon}{2}$ deletion fraction. Modify code to help locate blocks.

Idea: add constant-sized labels. Color blocks modulo 3. Use colors to guess and decode blocks.

Hope color boundaries same as block boundaries.

If placed enough correctly, get a lot of correct symbols. Decode if many windows are correct. Need to bound number of bad windows.

Analysis ideas: with $1-\varepsilon$ fraction of deletions, adversary can't affect too many blocks.

- Adversary can just delete enough information in block (must delete $>1-\frac{\varepsilon}{2}$ fraction)
- Delete separation from next same-colored neighbor.
- Wasted a lot of deletions to achieve same effect. (Could just delete info in the middle.) If number of colors is poly $\left(\frac{1}{\varepsilon}\right)$, then also expensive.

Open questions.

1. For binary codes, what is the highest fraction we can correct with constant rate. We don't know the answer existentially. It's in $[\frac{1}{3}, \frac{1}{2}]$.
2. How about fixed k?
3. Can we give efficient codes with better parameters?

7 Communications with partial noiseless feedback

Pritish Kamath

$C : B^k \to B^n$. Rate $\frac{k}{n}$.

We consider adversarial errors, and $|\Sigma| = 2$.

Upper boud $\frac{1}{4}$ (Plotkin) and lower bound $\frac{1}{4} - \varepsilon$ with rate $O(\varepsilon^2)$.

Berlekamp: communication with noiseless feedback. Bob tells Alice what he received. How many errors can you tolerate? Upper bound $\frac{1}{3}$, lower bound $\frac{1}{3} - \varepsilon$ with $O(\varepsilon)$.

Communication with partial noiseless feedback: what if Bob is allowed to send only δn bits of feedback for $\delta < 1$?

Coding for interactive communication, Schulman 1992. Generalization of Shannon's 1-way communication.

Alice and Bob engage in a protocol π. They want to make it robust to noise. Why is this not solved, why can't we encode each round? An adversary can completely corrupt 1 round.

What fraction of errors can $C(\pi)$ tolerate? He showed $\frac{1}{240}$. Braverman and Rao show an upper bound of $\frac{1}{4}$, and $\frac{1}{4} - \varepsilon$ can be achieved for large alphabet. For binary, $\frac{1}{8} - \varepsilon$ vs. upper bound $\frac{1}{6}$.

Between 0 and 1, $\frac{1}{4}$ and $\frac{1}{3}$. Randomized protocol, can correct up to $\frac{1}{3}$ for > 0.

Deterministic protocol: piecewise linear.

Communication with $\delta = 1$ feedback.

1. pad 1's after every bit to remove consecutive 0's. Repeat for N rounds.

2. Bob appends received bit, if transcript ends in 00, backtrack 3 bits. A: if Bob correct send next bit, else send 0. If $N > \frac{|y|}{3\varepsilon}$, this protocol tolerates $\frac{1}{3} - \varepsilon$ fraction errors. Rate $O(\varepsilon)$.

$\delta = \frac{2}{D}$-feedback

1. Alice sends D bits and Bob sends 2 bits. If Bob has correct so far, send next bit of y D times, else 0^D.

   Bob soft-decodes received bit: probabilistic decoding. Depending onhow many 1s, 1 or ?, or 0 or ?. Feedback is decoded bit.

Key insight: adversary might as well corrupt block entirely, or not corrupt it at all. Rate $O(\varepsilon\delta)$.

Deterministic communication with $\frac{2}{3}$ feedback. Derandomize. Pad 1's after every bit.

Soft-decode: decode as 0, 1 if number of 1's is 0 or 3, or ? else.

Open questions.

- What upper bounds can be shown on $f^{\text{det}}(\delta)$? Continuity at $\delta = 0$?

- What about rate vs. error tradeoff?

- Close the $\frac{1}{8}, \frac{1}{6}$ gap?

8 Fortification of Projection Games

Label cover game. Every edge in bipartite graph has a constraint $\pi_{xy} : \Sigma_X \to \Sigma_Y$. Accept if $\pi_{xy}(\sigma) = \sigma'$.

PCP Theorem: There exist constant sized alphabets such that it is NP-hard to check if a given label cover instance has value 1 or $\leq .999$.

Need a better gap to show NP-hardness reductions. Can we amplify the gap¿

Need polytime reduction: if $\text{val}(\mathcal{G}) = 1$ then $\text{val}(\mathcal{G}') = 1$; if $\leq .999$ then $\leq .001$.

Parallel repetition. Σ^k. If $\text{val}(\mathcal{G}) \leq \varepsilon_0$, is $\text{val}(\mathcal{G}^{\otimes k}) \leq \varepsilon_0^k$? This is false. However, the value does go down exponentially with k.

The state of the art is DS14: $\leq \left(\frac{2\sqrt{\varepsilon_0}}{1+\varepsilon_0}\right)^{\frac{k}{2}}$. Moshkovitz gives a method working for any constant gap.

Fortification: give necessary sufficient conditions fo this step to hold, and give optimal improvements.

Main idea. Q: why were previous proofs so hard? There is a smaller subgame of the repeated game, such that conditioned on it, the value in the kth coordinate game is large.

If the symmetrized game is robust, this won't happen.

Value of robust games goes down exponentially with k in parallel repetition.

Symmetrized games: check if labels of x and $\widetilde{x}$ project onto the same label of y.

$\text{val}(\mathcal{G}) \leq \text{val}(\mathcal{G}_{\text{sym}}) \leq \sqrt{\text{val}(\mathcal{G})}$.

A (δ, α)-robust game: For $S, T, |S|, |T| \geq \delta|\mathcal{X}|$, then $\text{val}(\mathcal{G}_{sym}|_{S\times T}) \leq \text{val}(\mathcal{G}_{sym}) + \alpha$. Concatentation makes a game robust.

Add bipartite on left and right. (Biregular). Play on last 2. Path.

What graphs can we concatenate? Constraining factor is degree D. Dependence of W's degree D on δ is crucial: alphabet size.

Necessary and sufficient conditions to make robust.

1. When does concat work? Easy case: If W complete graph, robust. Degree too large.

   (δ, α) extractor? We want $\|\mu_{ST} - \mathbf{u}\|_1 \leq \alpha$.

   What went wrong? Although both $\mu_S - u, \mu_T - u$ have small ℓ_1 norms, large ℓ_2 norms. If both have small ℓ_2 norms, then small.

   Bounded ℓ_2 norms suffice. Call such bipartite graphs fortifiers. $|S| \geq \delta|W|$ implies $\|\mu_S - u\|_1$... Do explicit fortifiers exist with small degree? Natural bipartite version of λ-spectral expander.

2. Fortifiers are necessary (when underlying game is far from being expander). $|\mu_S - u|_1 \leq \alpha$. $\|\mu_S\|^2 \leq \frac{O(\alpha)}{|X|}$.

3. Lower bounds on degree.

9 Separating DT from subcube partition complexity

Goal: compute $f : B^n \to B$ by reading as few input bits as possible. $D(f)$ deterministic query complexity. $R(f)$ randomized query complexity.

Techniques: block sensitivity, approximate degree, classical adversary, randomized certificate complexity... choose technique based on function.

New technique: Pb(f) partition bound subsumes all these techniques.

Also gives upper bound on query complexity:

$$Pb(f) \leq R(f) \leq Pb(f)^2.$$

Conjecture: $R(f) = \Theta(Pb(f))$. Aside: $Q(f) = \Theta(Qadv(f))$.

False! There is an asymptotic gap. There are f, $R(f) \geq 3.2^k$, $Pb(f) \leq 3^k$.

Intuition: Partition bound is too ambitious—bounds randomized subcube complexity. Open: Is $R^{sc}(f) = o(R(f))$? Main result: $R^{sc}(f) = o(R(f))$.

What is subcube partition complexity? Partition into f-monochromatic subcubes, each fixes d variables, $D^{sc}(f) = d$. Clearly $D^{sc}(f) \leq D(f)$.

Captain-crew function. $f(x_1, x_2, x_3, x_4) = x_1$ unless everyone else disagrees. $D(f) = 4$ (easy) and $D^{sc}(f) \leq 3$. How do we boost? Compose. D is multiplicative. $D(f^k) = 4^k, D^{sc}(f^k) \leq 3^k$.

$R^{sc}(f^k) \leq 3^k$. Difficult: $R(f^k) \geq (3.2)^k$.

Same lower bound techniques as 3MAJ. Guessing hard distribution is nontrivial due to asymmetry. Inductive argument is more involved.

What is best separation between $D(f), D^{sc}(f)$? $D \in [D^{sc}, D^{sc2}]$

Best separation between $R(f)$ and $R^{sc}(f)$? Similar bound. Comm complexity?

Devise lower bound techniques for $R(f)$ that do not also lower bound $R^{sc}(f)$.

10 Compression to entropy in protocols

Shorten conversation as much as possible while keeping its content.

External and internal information. $I^{ext}_\mu = I(M_\pi : XY)$. Number of bits an external observer learns on the input from the transcript.

Internal entropy is number of bits required to describe transcript to Alice plus ... to Bob.

$$H^{int}_\mu = H(M_\pi|X) + H(M_\pi|Y).$$

External simulation. External ε-error simulation of π if there exists dictionary such that distribution of (x, y, M_π) is ε-close in statistical distace to that of $(x, y, D(M_\sigma))$.

Internal simulation: $(x, y, M_\pi) \approx_\varepsilon (x, y, D_A(M_\sigma, x))$.

Braverman-Rao: internal information = amortized communication.

$H = I$ if no private randomness.

If 0-error simulation $CC_\mu(\sigma) \geq H_\mu(\pi)$.

If in π just Alice speaks, for all μ there is 0-error simulation $CC_\mu(\sigma) \leq H^{ext}_\mu(\pi) + 1$. (Huffman) In general, $CC_\mu(\sigma) \leq 2.18 H^{ext}_\mu(\pi) + 2$. Kushilevitz.

Internal compression upper bounds. BMY:

$$CC_\mu(\sigma) \leq O_\varepsilon(H^{int}_\mu(\pi)^2 \ln\ln CC_\mu(\pi)).$$

There exists private coin protocol. public coin simulation with info $O(k)$ has exponential loss in entropy.

Proof of Dietzfelbinger-Wunderlich. Simulate π with 0-error. Communicate 2 bits and convey 0.5 bits of information.

For every v in π, $R_v = \{(x, y) : v \text{ is ancestor of } M_\pi(x, y)\}$. Meaningful vertex: balanced or is lead and at least $\frac{2}{3}$ pass through it. For all μ, π there exists a meaningful vector. Go to bigger sum.

...

11 Correlation in Hard disributions in communication complexity

One-way model of communication: Alice to Bob.

Allow mixed strategies: let Alice and Bob use shared randomness.

Zero-sum game: computing $f(X, Y)$ means players win. Von Neumann's mimimax: optimal win rate is same for a worst case input and a hardest input distribution.

Is there a hardest inpt distribution that is product distribution? wr input partition into Alice/Bob's half. Or nearly-hardest?

Hard non-produce distribution resulting in complexity $\Omega(n)$, for Disj. $(X_i, Y_i) = (1, 1)$ with prob $\frac{1}{n}$, and equally shared among 3 other possibilities.

Optimally hard product distribution: $(1 - \frac{1}{\sqrt{n}})^2, \frac{1}{\sqrt{n}} - \frac{1}{n}, ..., \frac{1}{n}$. Complexity $\Omega(\sqrt{n})$.

BFS86: Disj requires $O(\sqrt{n} \ln n)$ communication under any product distribution. How about limited-correlaion distributions. How much correlation needed to prove $\Omega(n)$ bound. Quantum?

One way communication vs. PAC-learnability: Kremer, Nisan, Ron in 1999: for Boolean f,

$$R^{A \to B, I=0}_{\frac{1}{3}}(f) = \Theta(VC(f)).$$

VC-dimension characterizes PAC-learnability.

Is one-way communication under product distribution stronger than PAC-learnability?

classical optimal complexity $O(\sqrt{n})$, improving by $\ln n$.

Our protocol is based on previously known ideas, unlickly give optimal quantum (informational trace).

Design quantum protocol based on new principles, achieving complexity of $\widetilde{O}(n^{\frac{1}{4}})$. Classical $O(\frac{\sqrt{n(k+1)}}{\varepsilon^2})$. Quantum 4th, 3/2. Poly dependence on $\frac{1}{\varepsilon}$ is essential (no Chernoff).

Amount of correlation required for strong lower bounds. Exists: worst case complexity is $\Omega(n)$, but under bipartite dist with correlation less than $\frac{n}{1000}$ then dist complexity of f is $O(\ln n)$. "strongest possible counterexample."

PAC learnabilty is min length of example sequence that lets ε-learn g. Alice's input x defines $g_x(y) = f(x, y)$. For constant ε, PAC-complexity of $\{g_x : x \in B^n\}$ asymptotically equals the one-way communication cost of f¿

Alice plays teacher's role. Can she do more?

Error dependence of 1-way communication cost of f is $O(\ln\left(\frac{1}{\varepsilon}\right))$ while PAC $\Omega\left(\frac{1}{\varepsilon}\right)$.

Open

- Error dependence of optimal quantum protocol for Disj under product distributions? $O \ln\left(\frac{1}{\varepsilon}\right)$?
- Optimal error dependence of 2-way protocols under product distributions?
- Close remaining polylog gap in quantum communication complexity of Disj under distributions with limited correlation.

12 Multiparty pointer jumping

1. Generalized ER random graphs to handle edge dep
2. Prove bounds on clique num and chromatic num
3. improve NOF bounds for MPJ.

Concentration of measure quantify what most graphs look like.
B88: $\check{=}(1+o(1))\frac{-n\ln(1-p)}{2\ln n}$. BE76: clique number $r, r+1$, $r \approx \frac{2\ln n}{\ln\frac{1}{p}}$.
Threshold functions: connected. All monotone properties have threshold function (BT87).
Other results: graph evolution, degree sequence, eigenvalue distribution...
What if we allow edge dependencies?
$G_d(n,p)$: each edge depends on at most d other edges.
Ex. label vertices, add edge between same labels. Graph always has clique of size $\geq \frac{n}{2}$. Add edge between different. Bipartite.
Dependencies smaller in linear then get concentration of measure back.

1. If $\frac{d}{p} \ll \sqrt{n}$, then whp clique$(G) = \Omega\left(\frac{\ln n}{\ln\frac{1}{p}}\right)$. Same bound as ER up to constant!
2. $d \leq \frac{n}{(\ln n)^2}$: $O(d\ln n)$
3. $d = n^{o(1)}$: $\chi < \frac{-3n\ln(1-p)}{2\ln n}$.

Key intuition: let $G \sim G_p(n,p)$. Let $S \subseteq V$. Say $S \subseteq V$ uncorrelated if $G|_S$ independent.

Most small sets of vertices are uncorrelated. $dk^3 \leq n$. $\mathbb{P}(S \text{ is uncorrelated}) \geq 1 - \frac{3kd^3}{2n}$. Lots of ER graphs living inside. Patch ER result?

Classical clique LB (Bollobas88): $Y =$largest number of edge-disjoint k-cliques. Bound below by random process.

- select each k-clique independently with prob γ.
- remove all pairs of selected intersecting k-cliques
- L be set of remaining k-cliques.

$\mathbb{E}Y \geq \gamma\mathbb{E}\,(\text{k-cliques}) - 2\gamma^2\mathbb{E}\,(\text{intersecting k-cliques}) \geq \frac{n^2 p}{18k^5}$.

Edge martingale.

How make work for dependent?

Y is largest number of edge-disjoint uncorrelated k-cliques.

Add "uncorrelated" everywhere.

Independent case: $\mathbb{P}(S,T \text{ cliques}) = p^{\binom{k}{2}} p^{\binom{k}{2}-\binom{l}{2}}$.

Problem S, T can depend on each other. Dependent case $\mathbb{P}(S,T \text{ cliques}) < p^{\binom{k}{2}}$. Look at dependencies, quantify, technical stuff...

Martingale argument. $\mathbb{P}(Y=0) \leq \exp\left(-\frac{(\mathbb{E}Y)^2}{2\binom{n}{2}d^2}\right)$. Independent: learn info one bit at a time. increase by at most 1. In this case, expose new edge, could increase by d.

Everything is still small.

Communication complexity: $D(MPJ_3) = O\left(\frac{n(\ln\ln n)}{\ln n}\right)$. Pudlak, Rödl, Sgall prove for permutations. Modify PRS to work on al inputs. Correctness dependent on chromatic number bound for dependent random graphs.

13 Deterministically factoring sparse polynomials into multilinear factors and sms of univariate polynomials

Given a polynomial, find its irreducible factors. This is a natural problem with applications in list decoding...

It is einteger factorization: there exists a efficient randomized algorithm for factoring a polynomial given as a arithmetic circuit and as a black-box.

If f has circuit of size s then all it factors have circuits of size poly. No efficient deterministic algorithms known for poly factorizaiton.

Possible reason: poly factorization harder than PIT?

Other models? Look at sparsity of polynomial.

There exists deterministic identity testing for sparse polynomials. There exists efficient randomized algorithm for factoring sparse polynomials.

Problem: $\prod_{i=1}^n (x_i^n - 1) = \prod_{i=1}^n (1 + \cdots + x_i^{n-1}) \prod_{i=1}^n (x_i - 1)$. n^n dense factor of 2^n sparse polynomial: quasipolynomial blowup. GK85.

Easier problem: test sparse factorization. Given $m+1$ sparse polynomials $f, g_1, \ldots, g_m$ test $f = g_1 \cdots g_m$. (Restricted version of identity testing.)

Testing sparse factorization:

- symmetric case: when g_i equal.
- when g_i's are sums of univariate polynomials.
- factoring multilinear sparse polynomials
- multiquadratic sparse polynomials.

Addition sparse factorization/testing algorithm...?

Let $\mathcal{C}$ be class of polys. f is $\mathcal{C}$-split if f is product of polys from $\mathcal{C}$.

exist efficient deterministic factorization algorithms for multilinearly-split and sums-of-univariates-split sparse polynomials.

Both models have sparse factors: no representation problem. Extends SV10 (multilinear polys). Prior to work, no efficient deterministic factorization algorithms for any of these models.

Let $f = f_1 \cdots f_k \in \mathbb{F}[x_1, \ldots, x_n]$ be a poly of degree d. Factorization can be done in $d^{O(d)}$. Idea: reduce the number of variables.

Previous approach: construct map $H : \mathbb{F}^t \to \mathbb{F}^n$.

1. f_i is irreducible implying $f_i(H)$ is irreducible.
2. $f_i \nsim f_j \implies f_i(H) \nsim f_j(H)$.

Take H to be a random map. No known explicit constructions even for explicit classes.

Our approach: $H : \mathbb{F}^t \to \mathbb{F}^n$, $\Psi_H : \mathbb{F}^n \to \mathbb{F}^t$.

1. $\Psi_H(f_i)$ is an irreducible factor of $f_i(H)$.
2. $f_i \nsim f_j \implies \psi_H(f_i) \nsim \psi_H(f_j)$.
3. don't want mix: $\psi_H(f_j) | f_i(H) \implies \psi_H(f_i) \sim \psi_H(f_j)$.

Relaxation.

EFS for multilinear sparse polys and sums of univar polys.

(H, ψ_H) is an essential factorization scheme for $\mathcal{C}$ if given two irreducible polys, $\psi_H(f_1)$ is irred of $f_1(H)$. $\psi_H(f_1) \mid f_2(H)$ if $f_1 \sim f_2$.

If $f = \prod f_i, g = \prod g_i$ where $f_i, g_j \in \mathcal{C}$ irreducible, $f \equiv g$ iff $f(H) \equiv g(H)$.

$\prod f_i(H) = \prod g_i(H)$. $\psi_H(f_1) \mid f_1(H)$. Match $f_1 \sim g_j$, repeat.

New approach for poly factorization: EFS.

EFS for more classes of polynomials. Sparse polys with constant individual degree > 2.

Upper bound sparsity of factor.

Is quasi-poly worst possible blowup over char 0? Simplification: for perfect roots? (nth roots) Poly blowup for cubic?

13.1 A structure results for low degree poly and application

Low degree polys and subspace. $\mathbb{F} = \mathbb{F}_q$. Any $\mathbb{F}_q^n \to \mathbb{F}_q$ can be represented uniquely as multivar poly p with individual degrees $\leq q - 1$.

For f, let $K(f)$ be max dimension of $\mathbb{F}_q^n$ on which f is constant. $k_q(n, d) = \min\{K(f) : \deg f \leq d\}$. For any poly of deg d exists subspace of dim k where f is constant.

How does $k_q(n, d)$ behave?

- $d = 1 \implies k_q(n, d) = n - 1$.
- $d = 2, q = 2^m, k_q(n, d) = \frac{n}{2}$.

- Random degree d poly, $d_q(n,d) = O(dn^{\frac{1}{d-1}})$.
- Tardos Barringgon $k_2(n,d) = \Theta(dn^{\frac{1}{d-1}})$ for $d \leq \ln n$.
- Result $k_q(n,d) = \Theta(dn^{\frac{1}{d-1}})$.

Applications to pseudorandomness. Let f be a poly over n vars of degree $d \leq \ln n$. Then exists U of dim $\Omega(dn^{\frac{1}{d-1}})$ such that $f|_U$ =constant.

For $q = 2$, WLOG $f(0) = 0$. Grow subspace 1 at a time. Find basis for U step by step. Enough to show $U = \text{span}(\Delta_1, \ldots, \Delta_k)$, exists Δ_{k+1} such that $f|\ldots = 0$.

Maintiain cosets $x + U$ such that $f|_{x+U} = 0$.

$A_k = \{x \in \mathbb{F}_2^n : f|_{x+U} = 0\}$, if $|A_k| > 2^k$, can choose any $\Delta_{k+1} \in A_k \backslash U$, continue.

A_k expressed as set of nonzeros of small degree polys. enough to require $f(x+\sum_{i=1}^k a_i\Delta_i) = 0$ for all $a \in \mathbb{F}_2^k$ of hamming weight $\leq d$. Interpolate from values. $g(x) = \prod_{|a|\leq d}(1 - f(x + \sum_i a_i\Delta_i))$ then $x \in A_k$ iff $g(x) = 1$. g deg$\leq d(1 + \cdots + \binom{k}{d}) = O(k^d)$. By Schwartz-Zippel, if exists one solution to $g(x) = 1$ there are $\geq 2^{n-\deg(g)}$ solutions.

General fields: can define A_k and show corresponds to set of nonzeros of low degree poly $g(x)$. Wish find line in A_k. However, big sets with no line.

New direction $\mathbf{y}$, lin indep of $\Delta_{\leq k}$, $g(r \cdot \mathbf{y} \neq 0$ for all $r \in \mathbb{F}_q$. $h(y) = \sum_{r\in\mathbb{F}_q} g(r \cdot y)$. y good dir iff $H(y) \neq 0$. Trivial direction exists, so h nonzero. Many good directions exist.

Zeev Dvir to Kakeya set problem.

Affine extractor and disperser. Disperser: $f|_{u_0+U}$ not constant. Extracor: $|\mathbb{E}_{x\sim u_0+U}(-1)^{f(x)}| \leq \varepsilon$. Disperser known. Extractor: recent polylog. Degree d poly not $(n, o(dn^{\frac{1}{d-1}}))$ affine disperser/extractor. $AC^0[2]$ of depth 2 cannot compute good affine disperser/extractor. $AC^0[3]$ of depth 3 can. Affine extractors are variety extractors with related parameters. Low deg affine dis are affine extr. Affine extractor have small correlationswith low degree polys.

From affine disperser to affine extractor. Kaufman-Lovett 2008. (Bognadov-Viola.) (n,k) affine disperser of degree d, assume f is not (n, ck^{d-2}) affine extractor with error δ.

Generalization work for several polys simultaneously. Dimension k where constant.

14 Decomposing overcomplete 3rd order tensors using sum of squares

SVD $M = UDV^T = \sum_{i=1}^m d_i u_i v_i^T$. Matrix is sum of rank 1 matrices. We want the same thing for tensors. Rank 1 is $T_{i,j,k} = a(i)a(j)a(k)$. Low rank tensor decomposition.

Once we can do tensor decomposition, we can learn topic models, Mixture of Gaussians, HMM's, etc.

Tensor decomposition is unique: $I = RR^T$ for orthogonal R. (Rotation!) $T = \sum e_i^{\otimes 3}$ unique.

Tensor decomposition can be overcomplete: number of components $\gg$ number of dimensions.

Given tensor $T = \sum a_i^{\otimes 3}$ where $a_i \sim N(0, \frac{I}{n}) \in \mathbb{R}^n$, when $m < o(n^{1.5})$ our algorithm robustly find components $\{a_i\}$ with high probability in quasi-poly time. First algorithm for highly overcomplete 3rd order tensor decomposition.

Handle mildly overcomplete 3rd order tensors $m = O(n)$. Handle $m = \text{poly}(n)$ with at least 4th order tensor. Higher order tensors require more samples to accurately estimate.

Previous: Uniqueness for general position 3rd order tensor only when $2m \leq 3n - 2$.

Local convergence for tensors with randomlike components when $m < o(n^{1.5})$

How to find componets: Finding components $\iff$ optimizing polynomial.

Given tensor $T = \sum_{i=1}^{m} a_i \otimes a_i \otimes a_i$, for $x \in \mathbb{R}^n$ can compute poly $P(x) = \sum_{i=1}^{m} \langle a_i, x \rangle$, and when $m \leq o(n^{1.5})$, $P(x)$ only large when x close to a_i.

How can we optimize degree 3 polynomial?

Make the formulation convex: $\max \mathbb{E} P(x)$ such that X is distribution on unit sphere. Still cannot optimize. (too many distributions.) Replace distribution with pseudo-distribution: $\max \widetilde{\mathbb{E}} P(X)$ such that $\widetilde{\mathbb{E}}$ looks like expectation.

There is algorithm that optimizes over pseudo-distributions in $n^{O(r)}$ time, such that pseudo-distributions are indistinguishable to real distributions by polys of degree $\leq r$.

Example: bound degree 4 poly: $Q(x) = \sum_{i=1}^{m} \langle a_i, x \rangle^4$, $\|x\| = 1$. Flatten $M = \sum_{i=1}^{m} (a_i \otimes a_i)(a_i \otimes a_i)^T$. Bound spectral norm of M using low-degree proof.

3rd order tensor cannot be flattened symmetrically: $n \times n^2$.

Take the square of $P(x)$ to increase degree, flatten 4th order tensor, bound the spectral norm.

...

Way of flattening matters. $\sum_{i \neq j} (a_i \otimes a_j)(a_i \otimes a_j)^T$.

First quasi-poly time algorithm for 3rd order tensor decomposition in highly overcomplete case. Sum of squres useful in tensor decomp. Different ways of flatten lead to...

Improve running time to poly. Use SoS to bound 2 → 3 norm $P'(x) = \sum_{i=1}^{m} |\langle a_i, x \rangle^3|$. Decomposing overcomplete 3rd order tensors when components are not random.

Constant degree SoS can be viewed as SoS proof. Poly... $1 + o(1)$. When $x = a_i$, is $1 - o(1)$...

15 Dimension expanders via rank condensers

Michael Forbes

Theorem 15.1: New construction of explicit constant-degree dimension expanders over large fields. (Weakers results over small fields)

Idea: boolean pseudorandomness, sizes of subsets/min-entropy :: Linear algebraic pseudorandomness:dimensino of subspaces.

Prior work implicitly gave constructions.. Construction follow from developing connections in this theory to prior work.

Dimension expanders: $V \subseteq \mathbb{F}^n$ Consider $A, B, C : \mathbb{F}^n \to \mathbb{F}^n$. How affect dimension of V? $A(V), B(V)$. Constant number of maps, all small suspaces expand when take images together.

BISW04: $\mathbb{F}$ a field , $A_1, \ldots, A_d : \mathbb{F}^n \to \mathbb{F}^n$ is (ε, α)-dimension expander if for all $V \subseteq \mathbb{F}^n$ with $\dim V \leq \varepsilon n$, $\dim \operatorname{span}\{A_i(V)\}_i \geq \alpha \dim V$. Parameters

- degree d, ideally $d = O(1)$.

- expansion α, $\alpha \le d$. Ideally $\alpha \approx d$. Existentially.
- Goal: $(\frac{1}{2}, 1 + \delta)$-dimension expanders of constant degree.

Previous work:

1. (Wigderson2004, LubotzkyZelmanov08) irreducible representations of expander groups. Before, Cayley group. Char 0
2. DvirS, DW10. $A_i \in B^{n \times n}$ every $\mathbb{F}$. Via reduction to monotone vertex expanders. Monotone vertex expanders via expansion in $\mathrm{SL}_2(\mathbb{R})$.
3. $|\mathbb{F}| \ge \mathrm{poly}(n)$. Via linear algebra and polynomial method. Simpler than expander graphs. Can't get expander graph out of it. Better parameters.

Strategy

1. tensor: trivially obtain expansion. increase ambient space.
2. condense: reduce ambient space, roughly preserve expansion.
3. $\mathbb{F}^n$, dimension εn. Tensor degree d, $\mathbb{F}^{nd}$, dimension εdn. Condense degree $\approx d$: $\mathbb{F}^n, \varepsilon dn$.

Tensor is easy.

Condensing. Output smaller. $\mathbb{F}^t$ big enough to fit all inside of V. $E_1, E_2, \ldots$. How much dimension do we lose? For most maps preserve most of dimension...

QUantify: don't lose too much too often. 2 definitions.

Definition 15.2:

1. $\mathcal{E} : \{\mathbb{F}^n \to \mathbb{F}^t\}$ (r, ε)-lossy rank condenser if for all subspaces $V \subseteq \mathbb{F}^n$ with $\dim V = r$, some $E \in \mathcal{E}$ with $\dim E(V) \ge (1 - \varepsilon) \dim V$. (1 map that works. Small collection of maps.) (Method can give 99% that work.)
2. $\mathcal{E}$ (r, L)-lossless rank condenser if for all subspaces $V \subseteq \mathbb{F}^n$ with $\dim V = r$, $\sum_{E \in \mathcal{E}} (\dim V - \dim E(V)) \le L$. (Challenge: fix r, L, get as many maps as possible. Guarantee most maps have no loss.)

Need lossy. Implicit give lossless.

Lemma 15.3: $\mathcal{E}$ a (r, L) lossless condenser implies (r, ε)-lossy condenser with $|\mathcal{E}'| \le \frac{L}{\varepsilon r}$ if $|\mathcal{E}| > \frac{L}{\varepsilon r}$.

Guruswami Kopparty13: $|\mathbb{F}| \ge \mathrm{poly}(n)$. explicit $(r, \frac{nr}{t-r})$-lossless condenser. $|\mathcal{E}| > \frac{|F|}{t-r}$. Infinite number of maps, no loss for most! Originally phrased as subspace design.

Corollary: explicity (r, ε) lossy rank condenser $|\mathcal{E}| \le \frac{1}{n}\varepsilon(t - r)$.

$(n, \varepsilon n) \to (nd, \varepsilon dn) \xrightarrow{(\varepsilon dn, \delta)} (\mathbb{F}^n, (1 - \delta)\varepsilon dn)$.

Brute force, dim $\approx d^2$.

Cor. $|\mathbb{F}| \ge$ poly¿ Explicity $(\varepsilon, (1 - \delta)d)$ dimension expander of degree $\frac{d^2}{\delta(1 - \varepsilon d)}$.

Conclusions. Summary

- Previous ocnsructions used strong notions of spectral expansion
- dimension expanders emerging theory of linear-algebraic pr.
- construction: dimension expanders via tensoring and condensing.
- subspace designs of GK13 are lossless condensers, get lossy c
- connections with 2-source rank condensers, rank-metric codes, subspace evasive sets

Open; concat almost yields $O(\ln n)$-degree dim expanders over $\mathbb{F}_2$: improve? Our construction has expansion $\sqrt{d}$. Tensor plus condense, independent. Fold tensoring into condensing. Achieve $\Omega(d)$?

rank-preservingspaces.