Algorithmic Coding Theory Report

Soumyadeep Paul (BMC202178)
Madhav CS (BMC202132)

December 2023

This report is based on the paper **Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits** by Zeev Dvir and Amir Shpilka.

Summary

1 Locally decodable codes

Definition 1. *Let $\delta, \epsilon \in [0,1]$, and let q be an integer. We say that $E\colon \mathbb{F}^n \to \mathbb{F}^m$ is a (q, δ, ϵ)-locally decodable code if there exists a probabilistic oracle machine A such that*

- *in every invocation, A makes at most q queries (nonadaptively)*
- *for every $x \in \mathbb{F}^n$, for every $y \in \mathbb{F}^m$ with $\Delta(y, E(x)) < \delta m$, and for every $i \in [n]$, we have*

$$|\mathbb{F}| < \infty\colon Pr(A^y(i) = x_i) \geq \frac{1}{|\mathbb{F}|} + \epsilon$$

$$|\mathbb{F}| = \infty\colon Pr(A^y(i) = x_i) \geq \epsilon$$

Theorem 1. *Let $\delta, \epsilon \in [0,1]$, $\mathbb{F}$ be a field, and let $E\colon \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2, \delta, \epsilon)$-LDC. Then,*

$$m \geq 2^{\frac{\epsilon\delta n}{4} - 1}$$

2 $\Sigma\Pi\Sigma$ circuits

Definition 2. *Let $\mathbb{F}$ be a field. A $\Sigma\Pi\Sigma$ circuit, $\mathcal{C}$, over $\mathbb{F}$, with n inputs and k multiplication gates (i.e., top fan-in is k), is the formal expression*

$$\mathcal{C}(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d_i} L_{ij}(x)$$

where for each $i \in [k], j \in [d_i], L_{ij}$ is a nonconstant linear function,

$$L_{ij}(x) = L_{ij}^0 + L_{ij}^1 x_1 + \cdots + L_{ij}^n x_n$$

Definition 3 ($\Sigma\Pi\Sigma(k,d)$)**.** *Let $k, d > 0$ be integers. A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ is called a $\Sigma\Pi\Sigma(k,d)$ circuit if the following three conditions hold*

- *the top fan-in of $\mathcal{C}$ is k*
- *$d_1 = d_2 = \cdots = d_k = d$*
- *for every $i \in [k]$ and $j \in [d]$, L_{ij} is a homogeneous linear form, that is, $L_{ij}(x) = L_{ij}^1 x_1 + \cdots + L_{ij}^n x_n$ (The free coefficient in each linear functionis zero.)*

Lemma 1. *There exists a polynomial time algorithm such that, given as input a $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$, with top fan-in k and total degree $d > 0$, it outputs a $\Sigma\Pi\Sigma$(k, d) circuit $\mathcal{C}'$ such that $\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$. The circuit $\mathcal{C}'$ is called the corresponding $\Sigma\Pi\Sigma$(k, d) circuit of $\mathcal{C}$.*

Let $N_1, \cdots, N_k$ be the multiplication gates of $\mathcal{C}$. We define

$$gcd(\mathcal{C}) = gcd(N_1, \cdots, N_k)$$

Definition 4 (simple circuits)**.** *A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ is called simple if $gcd(\mathcal{C}) = 1$.*

For $\emptyset \neq T \subseteq [k]$ we define $\mathcal{C}_T$ as follows

$$\mathcal{C}_T(x) = \sum_{i \in T} c_i \prod_{j=1}^{d_i} L_{ij}(x) = \sum_{i \in T} c_i N_i(x)$$

Definition 5 (minimal circuits)**.** *Let $\mathcal{C} \equiv 0$ be a $\Sigma\Pi\Sigma$ circuit. We say that $\mathcal{C}$ is minimal if for every nonempty subset $T \subset [k]$, apart from $[k]$ itself, we have $\mathcal{C}_T \not\equiv 0$.*

Lemma 2. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, and let $\mathcal{C}'$ be the corresponding $\Sigma\Pi\Sigma(k,\ d)$ circuit. Then we have the following:*

- *rank($\mathcal{C}$) $\leq$ rank($\mathcal{C}$') $\leq$ rank($\mathcal{C}$) + 1.*
- *$\mathcal{C}$ is simple iff $\mathcal{C}'$ is simple.*
- *$\mathcal{C}$ is minimal iff $\mathcal{C}'$ is minimal.*

By the above lemma we can, WLOG, work with $\Sigma\Pi\Sigma(k,d)$ circuits.
Let $\pi\colon \mathbb{F}^n \to \mathbb{F}^n$ be an linear transformation. Let

$$\pi(\mathcal{C})(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d} \pi(L_{ij})(x)$$

We then have the following lemma.

Lemma 3. *Let π be a linear invertible transformation. Then,*

- *$\mathcal{C} \equiv 0$ iff $\pi(\mathcal{C}) \equiv 0$*
- *$\mathcal{C}$ is simple iff $\pi(\mathcal{C})$ is simple*
- *$\mathcal{C}$ is minimal iff $\pi(\mathcal{C})$ is minimal*
- *rank($\mathcal{C}$) = rank($\pi(\mathcal{C})$)*

3 $\Sigma\Pi\Sigma$ circuits and LDCs

In this section we will show the relation between LDCs and depth-3 circuits. By the previous section we can, WLOG, work in $\Sigma\Pi\Sigma(k,d)$ circuits instead of general circuits.

Theorem 2. *Let $k \geq 3, d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma(k,c)$ circuit on n inputs, over a field $\mathbb{F}$. Then, there exists a linear $(2, \frac{1}{12}, \frac{1}{4}) - LDC, E : \mathbb{F}^{n_1} \to \mathbb{F}^{n_2}$, with*

$$\frac{rank(\mathcal{C})}{P(k)log(d)^{k-3}} \leq n_1,\ \text{and } n_2 \leq kd \text{ where } P(k) = 2^{O(k^2)}.$$

4 Structural theorem for zero depth-3 circuits

In this section we will use Theorem 2 to prove a structural theorem for zero depth-3 circuits.

Theorem 3. *Let $\mathcal{C} \equiv 0$, be a $\Sigma\Pi\Sigma(k,d)$ circuit. Then, there exists a partition of $[k] : T_1, T_2, \cdots, T_s \subset [k]$ with the following properties:*

- $\mathcal{C} = \sum_{i=1}^{s} \mathcal{C}_{T_i} = \sum_{i=1}^{s} gcd(\mathcal{C}_{T_i}) \cdot sim(C_{T_i})$.
- *For all $i \in [s], sim(\mathcal{C}_{T_i}) \equiv 0$ and is simple and minimal.*
- *For all $i \in [s], rank(sim(\mathcal{C}_{T_i}) \leq 2^{O(k^2)} log(d)^{k-2}$.*

In other words, the theorem says that every zero $\Sigma\Pi\Sigma$ circuit can be broken down into zero subcircuits of low rank (ignoring the g.c.d.). This fact will be used in the next section, in which we present PIT algorithms for $\Sigma\Pi\Sigma$ circuits.

Theorem 3 is a consequence of the following lemma.

Lemma 4. *Let $k \geq 3, d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma(k,d)$ circuit, Then*

$$rank(\mathcal{C}) \leq 2^{O(k^2)} log(d)^{k-2}.$$

5 PIT Algorithms

Lemma 5. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit with $rank(\mathcal{C}) = r$. Then, there exists a polynomial time algorithm, transforming $\mathcal{C}$ into a $\Sigma\Pi\Sigma(k,d)$ circuit $\mathcal{C}'$ such that*

- $\mathcal{C} \equiv 0 \Leftrightarrow \mathcal{C}' \equiv 0$,
- *$\mathcal{C}'$ contains only r variables.*

Lemma 6. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit and let $r = rank(\mathcal{C})$, $s = size(\mathcal{C})$. Then we can check if $\mathcal{C} \equiv 0$*

1. *deterministically in time $poly(s)(r+d)^r$*
2. *probabilistically in time $poly(s+\frac{1}{\epsilon})$ using $r \cdot (log(d)+log(\frac{1}{\epsilon}))$ random bits, with error probability ϵ.*

Theorem 4. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit, $s = size(\mathcal{C})$. Then, Algoritm 1 will check if $\mathcal{C} \equiv 0$. Further, the algorithm will run in time $poly(s) \cdot exp\left(2^{O(k^2)} log(d)^{k-1}\right)$.*

Theorem 5. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit, $s = size(\mathcal{C})$. Then, Algoritm 2 will check if $\mathcal{C} \equiv 0$. Further, the algorithm will run in time $poly(s + \frac{2^k}{\epsilon})$, will use $2^{O(k^2)} log(d)^{k-1} log(\frac{1}{\epsilon})$ random bits and will make an error with probability less than ϵ.*

Algorithm 1: Deterministic Algorithm

Data: A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$.
for *every* $T \subset [k]$ **do**
  Compute $r_T = rank(sim(\mathcal{C}_T))$
  if $r_T \leq 2^{O(k^2)} log(d)^{k-2}$ **then**
    Check if $sim(\mathcal{C}_T) \equiv 0$ using part 1 of Lemma 6
  end
end
if *There exists a partition of* $[k]$*, such that for every set* $T \subset [k]$ *in the partition* $sim(\mathcal{C}_T) \equiv 0$, **then**
  accept
else
  reject
end

Algorithm 2: Probabilistic Algorithm

Data: A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$.
for *every* $T \subset [k]$ **do**
  Compute $r_T = rank(sim(\mathcal{C}_T))$
  if $r_T \leq 2^{O(k^2)} log(d)^{k-2}$ **then**
    Check if $sim(\mathcal{C}_T) \equiv 0$ using part 2 of Lemma 6
  end
end
if *There exists a partition of* $[k]$*, such that for every set* $T \subset [k]$ *in the partition* $sim(\mathcal{C}_T) \equiv 0$, **then**
  accept
else
  reject
end