List-decoding Reed-Muller Codes
Sudan et al, 1999

Daniel Shahaf
Tel-Aviv University
December 2008
1 Introduction

- Warm-up
- Unique-decoding
- Reed-Muller Codes
- Lines in $\mathbb{F}^m$
Let $\mathbb{F}$ be a field of size $q$, and fix some distinct $\alpha_1, \ldots, \alpha_n \in \mathbb{F}$.

- A Reed-Solomon code $C : \mathbb{F}^k \rightarrow \mathbb{F}^n$ maps a message $m = \langle m_1, \ldots, m_k \rangle \in \mathbb{F}^k$ to the codeword $w = \langle p_m(\alpha_1), \ldots, p_m(\alpha_n) \rangle \in \mathbb{F}^n$ consisting of the evaluations of the polynomial $p_m(x) = \sum_{i=0}^{k-1} m_i x^i$.

- Reed-Solomon codes are $[n, k, n - (k - 1)]_q$ codes.
Schwartz-Zippel Lemma

Let $S \subseteq \mathbb{F}$ and take some (non-zero) $m$-variate polynomial $Q$ of total degree $L$. Then $Q(x) \neq 0$ for at least $1 - \frac{L}{|S|}$ of the points in $S$.

**Proof**

- By induction on $m$.
- $m = 0, 1$ trivial.
- For $m > 2$: write
  
  $$Q(x_1, \ldots, x_m) = \sum_{i=0}^{t} x_1^i Q_i(x_2, \ldots, x_m)$$

  and note that
  
  $$\left(1 - \frac{L-t}{|S|}\right) \left(1 - \frac{t}{|S|}\right) \geq \left(1 - \frac{L}{|S|}\right).$$
Agreement

We will need a notion of distance:

**Definition**

The disagreement $\delta(w, w')$ of two codewords $w, w'$ is the fraction of coordinates $x$ wherein $w(x) \neq w'(x)$.

- It is a normalized Hamming distance.

**Definition**

The agreement $\tau(w, w')$ of two codewords $w, w'$ is the fraction of coordinates $x$ wherein $w(x) = w'(x)$. 
Consider a code \( C : \mathbb{F}^k \rightarrow \mathbb{F}^n \) with distance \( d \).

**Definition (unique-decoder)**

An algorithm \( A \) is said to correct \( e \) errors \( (e \leq \frac{d-1}{2}) \) if \( A(y) = x \in \mathbb{F}^n \) for every \( y \in \mathbb{F}^n \) whenever \( x \in \text{Im } C \) and \( d_H(x, y) \leq e \).
Decoders

Consider a code $C : \mathbb{F}^k \rightarrow \mathbb{F}^n$ with distance $d$.

**Definition (unique-decoder)**

An algorithm $A$ is said to **correct** $e$ errors ($e \leq \frac{d-1}{2}$) if $A(y) = x \in \mathbb{F}^n$ for every $y \in \mathbb{F}^n$ whenever $x \in \text{Im} \ C$ and $d_H(x, y) \leq e$.

**Definition (local unique-decoder)**

Similarly, an algorithm $A$ as above which also gets as input a coordinate $i \in [n]$ and outputs $x_i$ is said to be a **local unique-decoding algorithm**.

- $x_i$ is well-defined because $x = A(y)$ is well-defined.
Complexity

Polynomial time?

- Would like a unique-decoder to run in $\text{poly}(\text{output})$ time.
  - The output, a message, is shorter than the input (a codeword).
  - Sometimes, drastically shorter.

- Can beat the usual $\text{poly}(\text{input})$ due to the redundancy in the encoding.
Complexity

Polynomial time?

- Would like a unique-decoder to run in $\text{poly}(\text{output})$ time.
  - The output, a message, is shorter than the input (a codeword).
  - Sometimes, drastically shorter.
- Can beat the usual $\text{poly}(\text{input})$ due to the redundancy in the encoding.

Local-decoding

- For the local unique-decoder, we will permit the running time to depend on $|i| = \log n$ as well.
- Still, it’s expected to be faster than the global decoder.
Reed-Muller Codes

**Definition**

The \((m, d, q)\) Reed-Muller code maps tuples of \(\mathbb{F} = \text{GF}(q)\) to the evaluations of \(m\)-variate polynomials with total degree \(d\).

- **Parameters:** \([q^m, \binom{m+d}{d}, 1 - \frac{d}{q}]_q\).
- **Codewords are viewed as functions** \(w : \mathbb{F}^m \rightarrow \mathbb{F}\).
As special cases, the Reed-Muller code contains the Reed-Solomon code \((m = 1, d = k - 1)\) and Hadamard \((m = n, d = 1)\) codes.

Alternatively, we can take \(m = \log k\) and \(d = (\log k) / \log \log k = o(\log k)\) to get low-degree, low-variate polynomials (for message length \(k\)).
Lines in $\mathbb{F}^m$

**Definition**

We have two ways to define what a line in $\mathbb{F}^m$ is:

- A function mapping $t \in \mathbb{F}$ to $\ell_{a,b}(t) := a + tb \in \mathbb{F}^m$.
- A set of points of the form $\ell_{a,b} := \{ a + tb \mid t \in \mathbb{F} \}$. 
Properties of Lines

**Independence:**

- For fixed, distinct $t_1, t_2 \in \mathbb{F}$, and random $A, B$, the random variables $\ell_{A,B}(t_1)$ and $\ell_{A,B}(t_2)$ are distributed independently and uniformly on $\mathbb{F}^m$.

- For fixed $a \in \mathbb{F}^m$, non-zero $t \in \mathbb{F}$, and uniformly random $B$, the random variable $\ell_{a,B}(t)$ is distributed uniformly in $\mathbb{F}^m \setminus \{a\}$.

**Proof?**
Properties of Lines

**Independence:**

- For fixed, distinct $t_1, t_2 \in \mathbb{F}$, and random $A, B$, the random variables $\ell_{A,B}(t_1)$ and $\ell_{A,B}(t_2)$ are distributed independently and uniformly on $\mathbb{F}^m$.
- For fixed $a \in \mathbb{F}^m$, non-zero $t \in \mathbb{F}$, and uniformly random $B$, the random variable $\ell_{a,B}(t)$ is distributed uniformly in $\mathbb{F}^m \setminus \{ a \}$.

**Composition:**

- The composition $p|_\ell : \mathbb{F} \to \mathbb{F}$ of a line $\ell : \mathbb{F} \to \mathbb{F}^m$ and an $m$-variate polynomial $p : \mathbb{F}^m \to \mathbb{F}$ is a univariate polynomial of the same (total) degree as $p$.
- $p|_{\ell_{a,b}}(t) = p(a + tb)$. 

Proof?

Composition: 

The composition $p|_\ell : \mathbb{F} \to \mathbb{F}$ of a line $\ell : \mathbb{F} \to \mathbb{F}^m$ and an $m$-variate polynomial $p : \mathbb{F}^m \to \mathbb{F}$ is a univariate polynomial of the same (total) degree as $p$.

$p|_{\ell_{a,b}}(t) = p(a + tb)$. 

[s]
2 Local Unique-decoding

- The Simple Decoder
- The Improved Decoder
Idea: to find \( p(a) \), approximate \( p|_{\ell} \) for some line \( \ell = \ell_{a,b} \) through \( a \). (Note that \( p(a) = p|_{\ell_{a,b}}(0) \).)

Interface

- Input: Oracle access to noisy codeword \( f \), point \( a \in \mathbb{F}^m \), degree parameter \( d \).
- Promise: The distance of \( f \) from a valid codeword is at most \( \frac{1}{3(d+1)} \).
  - A valid codeword is evaluations of a degree-\( d \) polynomial.
- Output: \( p(a) \).
Definition

Code
- Pick random and independent $b \in \mathbb{F}^m$. Let $\alpha_1, \ldots, \alpha_{d+1} \in \mathbb{F}$ be non-zero field elements. Set $\beta_i = f(\ell_{a,b}(\alpha_i))$.
- Find a univariate polynomial $h$ that passes through $(\alpha_i, \beta_i)$ for all $1 \leq i \leq d + 1$.
  - $h$ approximates $p|\ell_{a,b}$.
- Print $h(0)$. 

Definition

Code
- Pick random and independent $b \in \mathbb{F}^m$. Let $\alpha_1, \ldots, \alpha_{d+1} \in \mathbb{F}$ be non-zero field elements. Set $\beta_i = f(\ell_{a,b}(\alpha_i))$.
- Find a univariate polynomial $h$ that passes through $(\alpha_i, \beta_i)$ for all $1 \leq i \leq d + 1$.
  - $h$ approximates $p|\ell_{a,b}$.
- Print $h(0)$.
Correctness

Lemma

For every \( a \in \mathbb{F}^m \), if \( \delta(f, p) \leq \frac{1}{3(d+1)} \), then the Simple Decoder outputs \( p(a) \) with probability \( 2/3 \).

Proof

Define events \( B_i = [\beta_i \neq p|_{\ell_{a,b}(\alpha_i)}] \).

- A fixed \( B_i \) occurs iff \( f(\ell_{a,b}(\alpha_i)) \neq p(\ell_{a,b}(\alpha_i)) \).
- Recall that the point \( \ell_{a,b}(\alpha_i) \) is random.
- Therefore, the probability of non-collision is exactly the average disagreement \( \delta := \delta(f, p) \).
- At probability \( 1 - (d + 1)\delta \geq 2/3 \), none of the \( B_i \)'s occur, and then \( h = p|_{\ell_{a,b}} \) and \( h(0) = p(\ell_{a,b}(0)) = p(a + 0b) = p(a) \).
Room for Improvement

- The error resistance was relatively low: $\Theta(1/d)$.
- The factor $1/(d + 1)$ caused by applying the union bound to $d + 1$ events.
  - We required $d + 1$ queries with no errors.
The error resistance was relatively low: $\Theta(1/d)$.

The factor $1/(d + 1)$ caused by applying the union bound to $d + 1$ events.

- We required $d + 1$ queries with no errors.

Trade-off: we’ll run more queries, but allow for some errors.

- The number of queries we set, arbitrarily, to $5(d + 1)$.

We’ll need an algorithm to recover a polynomial that agrees with most of the measured points.

- We can use the Berlekamp-Welch algorithm.
- The numbers check out.
Declaration

Idea: allow for some errors (events $B_i$) in the sampling; use Reed-Solomon unique-decoding to correct them.

Interface

- Input: Oracle access to noisy codeword $f$, point $a \in \mathbb{F}^m$, degree parameter $d$.
- Promise: The distance of $f$ from a valid codeword is at most $1/5$.
  - A valid codeword is evaluations of a degree-$d$ polynomial.
- Output: $p(a)$. 
Definition

Code

- Pick random and independent $b \in \mathbb{F}^m$, and non-zero $\alpha_1, \ldots, \alpha_{5(d+1)} \in \mathbb{F}$. Set $\beta_i = f(\ell_{a,b}(\alpha_i))$.

- Find a univariate polynomial $h$ that passes through at least 60% of the points $(\alpha_i, \beta_i)$ as $1 \leq i \leq 5(d + 1)$.
  - $h$ approximates $p|_{\ell_{a,b}}$.

- Print $h(0)$. 
Correctness

Lemma

For every \( a \in \mathbb{F}^m \), if \( \delta(f, p) < 1/5 \), then the Improved Decoder outputs \( p(a) \) with probability \( > 1/2 \).

Proof

Define events \( B_i = [\beta_i \neq p|\ell(\alpha_i)] \).

- As before, the probability of \( B_i \) is exactly \( \delta := \delta(f, p) \).
- Define an indicator variable \( B' = [\sum B_i > 2(d + 1)] \).
- By Markov’s Inequality,
  \[
  \Pr[B'] = \Pr[\sum_{i=1}^{5(d+1)} B_i > 2(d + 1)] < \frac{5\delta}{2} \cdot \frac{d + 1}{d + 1}.
  \]
- Require \( 5\delta/2 < 1/2 \).
Local List-decoding

- Definitions
- The Algorithm
- Theoretical Foundation
Global List-decoding

**Definition**

The \( \tau \)-agreement of \( f \in \mathbb{F}^m \to \mathbb{F} \), denoted \( \text{Ag} (f, \tau) \), is the set of all functions \( g \in \mathbb{F}^m \to \mathbb{F} \) that agree with \( f \) on (at least) \( \tau \) coordinates.

- \( \tau \) may be the absolute number or a fraction.

Caveat: \( A = \lambda f. (a_1, \ldots, a_{|\mathbb{F}^m|}) \) meets the definition. But runs in super-polynomial time.

How to extend to local decoding?
Global List-decoding

Definition

The $\tau$-agreement of $f \in \mathbb{F}^m \rightarrow \mathbb{F}$, denoted $\text{Ag}(f, \tau)$, is the set of all functions $g \in \mathbb{F}^m \rightarrow \mathbb{F}$ that agree with $f$ on (at least) $\tau$ coordinates.

- $\tau$ may be the absolute number or a fraction.

Definition

An algorithm $A$ is said to be a $(\tau, L)$ list-decoder if for every noisy codeword $f \in \mathbb{F}^m \rightarrow \mathbb{F}$ it outputs an $L < L'$-sequence $(z_1, \ldots, z_{L'}) \in (\mathbb{F}^m \rightarrow \mathbb{F})^{L'}$ that contains each legitimate codeword $y \in \text{Ag}(f, \tau)$ at least once.

Global List-decoding

Definition

The $\tau$-agreement of $f \in \mathbb{F}^m \rightarrow \mathbb{F}$, denoted $\text{Ag}(f, \tau)$, is the set of all functions $g \in \mathbb{F}^m \rightarrow \mathbb{F}$ that agree with $f$ on (at least) $\tau$ coordinates.

- $\tau$ may be the absolute number or a fraction.

Definition

An algorithm $A$ is said to be a $(\tau, L)$ list-decoder if for every noisy codeword $f \in \mathbb{F}^m \rightarrow \mathbb{F}$ it outputs an $L < L'$-sequence $(z_1, \ldots, z_L') \in (\mathbb{F}^m \rightarrow \mathbb{F})^{L'}$ that contains each legitimate codeword $y \in \text{Ag}(f, \tau)$ at least once.

- Caveat: $A = \lambda f. (a_1, \ldots, a_{|\mathbb{F}^m \rightarrow \mathbb{F}|})$ meets the definition.
  - But runs in super-polynomial time.
- How to extend to local decoding?
Local List-decoding

Definition

An algorithm $A$ is said to be a $(\tau, L)$ local list-decoder if ...

.$$...?$$
Local List-decoding

Definition

An algorithm $A$ is said to be a $(\tau, L)$ local list-decoder if there exist $L' > L$ algorithms $A_i = A(i)$ (for $1 \leq i \leq L'$) such that, for every noisy codeword $f \in \mathbb{F}^m \rightarrow \mathbb{F}$ and for every codeword $p \in \text{Ag}(f, \tau)$, there exists an $s \in [L']$ such that, for every $a \in \mathbb{F}^m$, the probability (over coin tosses of $A_s$) that $A_s$ computes $p(a)$ is at least $2/3$:

$$A \in \text{LLD}_{\tau, L} \iff \exists L' > L : \exists A_1, \ldots, A_{L'} :$$
$$\forall f \in \mathbb{F}^m : \forall p \in \text{Ag}(f, \tau) :$$
$$\exists s \in [L'] : \forall a \in \mathbb{F}^m :$$
$$\text{Prob}_{A_s}[A_s(a) = p(a)] \geq 2/3.$$
Overview

- Goal: local list-decoding of the Reed-Muller code.
- We will consider polynomials with very low agreement with $f$.
  - We will have to account for this in the interpolation step.
  - Use Reed-Solomon list-decoding.
- Each such polynomial will be uniquely specified by an advice associated with it.
  - The advice will be its evaluation at a point.
Helper Function

We use the following family of subroutines $A_{z,\gamma}(a,f)$.

**Amplifier**($a, f$)

- **Input:** Oracle access to noisy codeword $f$, point $a \in \mathbb{F}^m$, degree parameter $d$, agreement parameter $\tau$.
- **Hard-wired:** point $z \in \mathbb{F}^m$ and advice $\gamma \in \mathbb{F}$.
Helper Function

We use the following family of subroutines $A_{z,\gamma}(a, f)$.

**Amplifier($a, f$)**

- **Input:** Oracle access to noisy codeword $f$, point $a \in \mathbb{F}^m$, degree parameter $d$, agreement parameter parameter $\tau$.
- **Hard-wired:** point $z \in \mathbb{F}^m$ and advice $\gamma \in \mathbb{F}$.
- Take the line $\ell = \{ a + t(z - a) \mid t \in \mathbb{F} \}$.
- Find all degree-$d$ univariate polynomials $h_1, \ldots, h_n$ that have relative agreement $\tau/2$ with $f|\ell$ (over $\mathbb{F}$).
  - $h_i$ approximate $p|\ell$.
- If there is a **unique** $h_i$ that evaluates to $\gamma$ at $z$, print $h_i(0) \approx p(a)$. 

Local List-decoder

This is the main subroutine of the local list-decoder algorithm $A_{\text{LLD}}$.

**Code for $A_{\text{LLD}}$**

- **Input**: Oracle access to noisy codeword $f$, point $a \in \mathbb{F}^m$, degree parameter $d$, agreement parameter parameter $\tau$.

- For each $(z, \gamma) \in \mathbb{F}^m \times \mathbb{F}$ independently:
  - Run the unique-decoder with $A_{z,\gamma}$ as oracle.
  - Print whatever it does (if any).
We will show:

- That $A_{z,p(z)} = p(a)$ with arbitrarily large probability over random choices of $a$ and $z$.
- That all polynomials with relative agreement $\tau/2$ are pairwise distinct on a large fraction of the space.
Correctness

We will show:

- That $A_{z,p(z)} = p(a)$ with \textit{arbitrarily large} probability over random choices of $a$ and $z$.
- That all polynomials with relative agreement $\tau/2$ are pairwise distinct on a large fraction of the space.

To see that the algorithm $A_{\text{LLD}}$ is \textbf{correct}:

- There is a \textit{fixed} $z \in \mathbb{F}^m$ such that
  \[ \text{Prob}_{a \in \mathbb{F}^m} [A_{z,p(z)}(a) = p(a)] > 1 - \epsilon. \]
- When $A_{\text{LLD}}$ tries that $z$ and $\gamma = p(z)$, then $A_{z,\gamma} = A_{z,p(z)}$ will be correct over $1 - \epsilon$ of all $a$'s.
- Therefore, the unique-decoder will compute $p(a)$ correctly at a probability of $2/3$. 
Time Analysis

Proposition

If $\tau > 2\sqrt{d/q}$ then $A_{z,\gamma}$ runs in $\text{poly}(q, m)$ time.

Proof

- Need to show we meet the Reed-Solomon list-decoding precondition, which is $t > 2\sqrt{kn}$ for finding all degree-$k$ polynomials that agree on at least $t$ points out of $n$.
- $t = \tau q/2$, $n = q$, $k = d$. 
Proposition

Let \( p_1, \ldots, p_n \) be all \( m \)-variate degree-\( d \) polynomials over \( \mathbb{F} \) having agreement \( \tau \geq \sqrt{2d/q} \) with \( f \). Then:

- \( n \leq 2/\tau \).
- The probability (over uniform choice of \( z \in \mathbb{F}^m \)) that all \( p_i(z) \) are distinct is at least \( 1 - \frac{2d}{\tau^2 q} \).
Proposition

Let $p_1, \ldots, p_n$ be all $m$-variate degree-$d$ polynomials over $\mathbb{F}$ having agreement $\tau \geq \sqrt{2d/q}$ with $f$. Then:

- $n \leq 2/\tau$.
- The probability (over uniform choice of $z \in \mathbb{F}^m$) that all $p_i(z)$ are distinct is at least $1 - \frac{2d}{\tau^2 q}$.

Proof

For the second part, let $B_{ij} = [p_i(z) = p_j(z)]$ for $i < j$. Note that $\text{Prob}[B_{ij}] \leq d/q$ and apply the union bound to get $1 - \binom{n}{2} \frac{d}{q}$.
Proposition

Let \( p_1, \ldots, p_n \) be all \( m \)-variate degree-\( d \) polynomials over \( \mathbb{F} \) having agreement \( \tau \geq \sqrt{2d/q} \) with \( f \). Then:

- \( n \leq 2/\tau \).
- The probability (over uniform choice of \( z \in \mathbb{F}^m \)) that all \( p_i(z) \) are distinct is at least \( 1 - \frac{2d}{\tau^2q} \).

Proof

For the second part, let \( B_{ij} = [p_i(z) = p_j(z)] \) for \( i < j \). Note that \( \text{Prob}[B_{ij}] \leq d/q \) and apply the union bound to get \( 1 - \binom{n}{2} \frac{d}{q} \).

Conclusion

With high probability, a codeword \( w = p \in \mathbb{F}^m \rightarrow \mathbb{F} \) is uniquely defined by its value on a random point.
Lemma

Let $p$ be a polynomial that has $\tau$-agreement with $f$. Take some $\epsilon > 0$ such that $q \geq \frac{16(d + 1)}{\tau^2 \epsilon}$. Then the probability over random $\alpha, z \in \mathbb{F}^m$ that $A_{\alpha, p(z)}$ prints $p(\alpha)$ is at least $1 - \epsilon$. 
The Amplifier Amplifies

Lemma

Let $p$ be a polynomial that has $\tau$-agreement with $f$. Take some $\epsilon > 0$ such that $q \geq 16(d + 1)/\tau^2\epsilon$. Then the probability over random $a, z \in \mathbb{F}^m$ that $A_{z,p}(z)$ prints $p(a)$ is at least $1 - \epsilon$.

Proof overview

- Define events:
  - $B$ is “the agreement of $p$ and $f$ on $\ell$ is less than $\tau/2$.”
  - $C$ is “there exist two univariate polynomials, each having $\tau/2$ agreement with $f|_{\ell}$, that evaluate equally at $z$.”
- Convince: if neither occurs, then $A_{z,p}(z)$ prints $p(a)$.
- Bound the probabilities of $B$ or $C$ occurring.
Bounding the Probabilities of “Bad” Events

Claim

\[ \text{Prob}_{z,a}[B] \leq 4/\tau q. \]

Proof

Let \( B_t \) be the indicator variable of “agreement at a given point \( \ell(t) \in \ell \) (for \( t \in \mathbb{F} \))”. Its probability is \( \tau \). Use Chebyshev’s inequality on the random variable \( \sum B_t \).

Claim

\[ \text{Prob}_{z,a}[C] \leq 8d/\tau^2 q \text{ whenever } \tau > 2\sqrt{d/q}. \]

Proof

Use the previous proposition with \( \tau := \tau/2 \).
To finish the lemma, pick large enough $q$ so each event has probability $< \epsilon/2$.

$q \geq 16(d+1)/\tau^2 \epsilon$ is sufficient.

The probability that neither $B$ nor $C$ would occur is $1 - \epsilon$. 

Summary

1. Introduction
2. Local Unique-decoding
3. Local List-decoding
Noise proves nothing.
—Mark Twain

The End.