**6.440 Essential Coding Theory** March 19, 2008

# Lecture 13

*Lecturer: Madhu Sudan* *Scribe: Ankur Moitra*

## 1 Overview

Last lecture we proved the Johnson Bound using a combinatorial argument, and in this lecture we will demonstrate how to algorithmically list-decode Reed-Solomon Codes and achieve the Johnson Bound.

## 2 The Johnson Bound

In the last lecture we proved that if a code $C$ is an $(n, k, d)_q$ code, then this code is also $(p, L)$ list decodable for

$$p = 1 - \sqrt{1 - \frac{d}{n}}$$

and $L$ is a fixed polynomial in $n$. This is known as the Johnson Bound. An equivalent condition is that for all $\mathbf{y} \in \{0, 1\}^n$, there are at most $L = poly(n)$ codewords $c \in C$ contained in the ball of radius $pn$ centered around $\mathbf{y}$. Then a received message $\mathbf{y}$ can be list decoded by outputting all codewords contained in $B(\mathbf{y}, pn)$.

However the combinatorial proof of the Johnson Bound did not yield an efficient algorithm for performing list decoding. The only algorithm that can be recovered from this proof is to brute force list decode by enumerating all possible codewords, and outputting all the codewords contained in the ball $B(y, pn)$. In this lecture we will see that the Johnson Bound can be achieved for Reed-Solomon Codes.

An important note is that the Johnson Bound is an existential result. And there are $(n, k, d)_q$ codes that can achieve a larger list-decoding radius (for $L = poly(n)$) than the radius guaranteed by the Johnson Bound. We will see examples of these codes later in this class.

## 3 List Decoding Reed-Solomon Codes

Reed-Solomon Codes are $(n, k, n - k + 1)_q$ codes for $q$ is a prime power and $q \geq n$. Then the Johnson Bound implies that these codes can be

$$(1 - \sqrt{\frac{k-1}{n}}, L)$$

list decoded. Asymptotically, this implies that if there are $t > \sqrt{kn}$ indices that agree with the original codeword transmitted, that we can find at most $L$ codewords such that the original codeword is contained in this list.

The problem is to find this list algorithmically, given the received message. We will work towards this goal, achieving better and better guarantees for the list decoding radius until the Johnson Bound is achieved.

# 4 An Algebraic Representation

Suppose that the vector $\mathbf{y}$ is received. $y_i$ is the evaluation (possibly corrupted by noise) of the polynomial on the symbol $\alpha_i$. The first goal in list-decoding Reed-Solomon Codes is to express the received symbols algebraically as a low-degree polynomial rather than as a set of points.

**Lemma 1** *For all sets of $n$ points $\{\alpha_i, y_i\}_1^n$, there exists a bivariate polynomial $Q$ which is not identically zero such that $deg_x(Q), deg_y(Q) \leq \sqrt{n}$ and such that $Q(\alpha_i, y_i) = 0$ for all $i$.*

**Proof** Represent such a low-degree polynomial $Q$ as

$$Q(x, y) = \Sigma_{0 \leq i,j \leq \sqrt{n}} q_{i,j} x^i y^j$$

There are $(\sqrt{n}+1)^2 > n$ total coefficients, which are free variables. Now consider the constraints that $Q(\alpha_i, y_i) = 0$ for all $i$. Each such constraint is a homogenous linear constraint:

$$\Sigma_{0 \leq i,j \leq \sqrt{n}} q_{i,j} \alpha_l^i y_l^j = 0 \forall_l$$

These constraints are homogenous, and consequently the linear system is consistent. There are exactly $n$ linear constraints, and because there are strictly more free variables, this guarantees the existence of a non-trivial (not all $q_{i,j}$ are zero) solution. ■

This proof actually implies that we can find such a low degree polynomial $Q$ in polynomial time, because the coefficients can be found by solving a system of linear equations.

# 5 Using the Algebraic Representation

We will factor the low degree polynomial $Q$ to find all codewords that agree in at least $t$ indices with the received message.

**Lemma 2** *Suppose $Q$ is a non-trivial, bivariate polynomial such that $deg_x(Q) \leq D$, $deg_y(Q) \leq D$ and that $Q(\alpha_i, y_i) = 0 \forall_i$. Also assume that the pairs $(\alpha_i, y_i)$ are distinct. Then for any univarite polynomial $p$ that has degree at most $k$ and when evaluated on $\alpha$ agrees with $\mathbf{y}$ on at least $t > (k+1)D$ indices, $y - p(x)$ divides $Q(x, y)$*

**Proof** Consider $Q_x(y) = Q(x, y)$, a polynomial in $y$ such that coefficients are in the polynomial ring $F[x]$. Then $Q_x(y) \in (F[x])[y]$. Using the Division Algorithm, $y - \beta$ (where $\beta = p(x) \in F[x]$) divides $Q_x(y)$ iff $Q_x(\beta) = 0$.

Then define $g(x) = Q_x(p(x) = Q(x, p(x)) \in F[x]$. The term $x^D y^D$ will be the dominating term in calculating an upper bound for the degree of $g(x)$. $deg(g(x)) \leq D + Dk = (k + 1)D$.

Now consider any point $\alpha_i$ such that $p(\alpha_i)$ agrees with $y_i$. For such a point, $g(\alpha_i) = Q(\alpha_i, p(\alpha_i)) = Q(\alpha_i, y_i) = 0$. By assumption, there are at least $t$ such points and these $\alpha_i$ must be distinct because the pairs $(\alpha_i, y_i)$ are distinct by assumption, and the polynomial $p$ cannot evaluate to both $y_i$ and $y_j$ on $\alpha_i = \alpha_j$. If $t > (k + 1)D$, also by assumption, then this implies that $g(x) = 0$, and this yields the desired claim. ■

# 6 An Algorithm for List Decoding Reed-Solomon Codes

These lemmas yield a first algorithm for list decoding Reed-Solomon Codes:

Find $Q \neq 0$ such that $deg_x(Q), deg_y(Q) \leq \sqrt{n}$ and $Q(\alpha_i, y_i) = 0 \forall_i$

Factor $Q$, and report all polynomials $p$ such that $y - p(x)$ divides $Q$

A number of classical results yield algorithms that can factor bivariate polynomials in polynomial time. See the References section.

Our first lemma implies that such a bivariate polynomial $Q$ exists, and can be found in polynomial time. Our second lemma implies that we can find all polynomials $p$ such that $p$ agrees with $\mathbf{y}$ on $t > (k+1)\sqrt{n}$ indices, in polynomial time - because for all such polynomials $p$, $y - p(x)$ will divide $Q$ and $y - p(x)$ is irreducible, and consequently will be contained in our factorized list for $Q$.

The size of the list that we output will clearly be polynomial in $n$. In fact the list will be size at most $\sqrt{n}$ because $deg_y(Q) \leq \sqrt{n}$. And if at most $n-(k+1)\sqrt{n}$ errors occured, then the original codeword (and corresponding polynomial) will agree with $\mathbf{y}$ in at least $(k + 1)\sqrt{n}$ indices and will be output by our algorithm.

# 7 Optimizing Parameters

In our first algorithm, we chose a bivariate polynomial $Q$ that minimized the degree in $x$ and the degree in $y$ equally. This yields a trivial algorithm if $k \geq \sqrt{n}$, and in this section we will optimze parameters to get closer to the Johnson Bound.

**Claim 3** *For all sets of $n$ points $\{\alpha_i, y_i\}_1^n$, and for any $D_x, D_y$ such that $(D_x + 1)(D_y + 1) > n$, there exists a bivariate polynomial $Q$ which is not identically zero, $deg_x(Q) = D_x$, $deg_y(Q) = D_y$, and such that $Q(\alpha_i, y_i) = 0$ for all $i$.*

The proof follows immediately from the proof of our first lemma, and is again constructive because such a polynomial can be found in polynomial time.

**Claim 4** *Suppose $Q$ is a non-trivial, bivariate polynomial such that $deg_x(Q) \leq D_x$, $deg_y(Q) \leq D_y$ and that $Q(\alpha_i, y_i) = 0 \forall_i$. Also assume that the pairs $(\alpha_i, y_i)$ are distinct. Then for any univarite polynomial $p$ that has degree at most $k$ and when evaluated on $\alpha$ agrees with $\mathbf{y}$ on at least $t > D_x + kD_y$ indices, $y - p(x)$ divides $Q(x, y)$*

Again, this proof follows immediately from the proof of our second lemma. The only change is that the bound on $deg(g(x))$ is $deg(g(x)) \leq D_x + kD_y$.

Then we can choose $D_x = \sqrt{nk}, D_y = \sqrt{\frac{n}{k}}$ and this yields an algorithm that recovers all polynomials $p(x)$ that agree with $\mathbf{y}$, the received message, on at least $2\sqrt{kn}$ indices.

The above optimization in parameters relied on jointly choosing $deg_x(Q)$ and $deg_y(Q)$ to minimize $deg_x(Q) + kdeg_y(Q)$, while still guaranteeing the existence of a bivariate polynomial that satisfies the required conditions. However, we can improve this by realizing that the polynomials $x^i y^j$ and $x^i + y^j$ have identical degrees in each variable, but the second polynomial yields a much smaller bound on the degree of $g(x) = Q(x, p(x))$. The real cost for any term $x^i y^j$ is $i + kj$, and the total cost for $Q$ is the maximum cost. So we can optimize our choice of monomials in $Q$ by choosing greedily until at least $n + 1$ monomials are in $Q$, and then choosing the coefficients for these monomials to fit our polynomial to the constraints.

**Theorem 5 (Sudan 97)** *There exists a polynomial time algorithm to list decode Reed-Solomon Codes from $t > \sqrt{2kn}$ agreement*

**Claim 6** *For all sets of $n$ points $\{\alpha_i, y_i\}_1^n$, and for any $C = \{q_{i,j}\}$ such that $|C| > n$, there exists a bivariate polynomial $Q(x, y) = \Sigma_{i,j} q_{i,j} x^i y^j$ which is not identically zero, $q_{i,j} \neq 0 \Rightarrow q_{i,j} \in C$, and such that $Q(\alpha_i, y_i) = 0$ for all $i$.*

**Claim 7** *Suppose $Q(x, y) = \Sigma_{i,j} q_{i,j} x^i y^j$ is a non-trivial, bivariate polynomial such that $q_{i,j} \neq 0 \Rightarrow i + kj \leq D$ and $Q(\alpha_i, y_i) = 0 \forall_i$. Also assume that the pairs $(\alpha_i, y_i)$ are distinct. Then for any univarite polynomial $p$ that has degree at most $k$ and when evaluated on $\alpha$ agrees with $\mathbf{y}$ on at least $t > D$ indices, $y - p(x)$ divides $Q(x, y)$*

Choose all pairs $i, j$ such that $i + kj \leq \sqrt{2kn}$. These are the integer points contained in the triangle bounded by the $x$-axis, $y$-axis and the line $x + ky = \sqrt{2kn}$. There are more than $\frac{1}{2} \times \sqrt{\frac{2n}{k}} \times \sqrt{2kn} = n$ such points. This yields an algorithm that recovers all polynomials $p(x)$ that agree with $\mathbf{y}$, the received message, on at least $\sqrt{2kn}$ indices. And this yields a proof of the theorem.

# 8 Reaching the Johnson Bound

In this section we will reach the Johnson Bound for list decoding Reed-Solomon Codes. We will do this by finding a higher degree bivariate polynomial $Q$ such that $Q$ has a multiplicity $m$ zero point at $\alpha_i, y_i$ for all $i$.

**Theorem 8 (Guruswami, Sudan 98)** *For any $t > \sqrt{kn}$ there exits a polynomial time algorithm to list decode Reed-Solomon Codes from $t$ agreement.*

Define a polynomial $Q(\bar{x}, y) = Q(x + \alpha_i, y + y_i)$. Then $Q(\bar{x}, y)$ has a multiplicity $m$ zero at $0, 0$ iff $Q(x, y)$ has no support on monomials of total degree $\leq m - 1$. Then the constraint that $Q(\bar{x}, y)$ has a multiplicity$m$ zero at $0, 0$ implies that for all coefficients $q_{\bar{i},j}$ such that $i + j \leq m - 1$ are zero.

**Claim 9** *There are $\binom{m+1}{2}$ pairs $(i, j)$ such that $i, j \geq 0$ and $i + j \leq m - 1$.*

Then each point $\alpha_i, y_i$ imposes $\binom{m+1}{2}$ linear constraints on $Q(\bar{x}, y)$ which are also linear constraints on $Q(x, y)$. Then there are $\binom{m+1}{2}$ $\times n$ total linear constraints, and we can choose $D = (m+1)\sqrt{nk}$ such that the triangle bounded by the $x$-axis, the $y$-axis and the line $x + yk \leq D$ contains at least

$$\frac{1}{2} \times (m+1)\sqrt{nk} \times \frac{m+1}{k}\sqrt{nk} > \binom{m+1}{2} n$$

points $(i, j)$ such that $i + kj \leq D$.

**Claim 10** *Suppose $Q(x, y) = \Sigma_{i,j} q_{i,j} x^i y^j$ is a non-trivial, bivariate polynomial such that $q_{i,j} \neq 0 \Rightarrow i + kj \leq D$ and $Q(\alpha_i, y_i) = 0$ with multiplicity $m$ for all $i$. Also assume that the pairs $(\alpha_i, y_i)$ are distinct. Then for any univarite polynomial $p$ that has degree at most $k$ and when evaluated on $\alpha$ agrees with $\mathbf{y}$ on at least $t > \frac{D}{m}$ indices, $y - p(x)$ divides $Q(x, y)$*

The proof for this follows immediately from the proof of our first lemma, because each zero is counted with multiplicity $m$.

This yields the Johnson Bound, because $D = (m + 1)\sqrt{nk}$ and for each $i$, $Q(\alpha_i, y_i) = 0$ with multiplicity $m$. Then we can recover all codewords that agree on $t > \frac{m+1}{m}\sqrt{nk}$ indices in polynomial time. And as $\lim_{m\to\infty}$ this achieves the Johnson Bound, and all codewords that agree on $t > \sqrt{kn}$ can be recovered. And this yields a proof of the theorem

# References


[1] V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon codes and algebraic-geometric codes. *Foundations of Computer Science, 1998. Proceedings.39th Annual Symposium* , vol. 45(6), pp. 28-37, 1998.

[2] A. K. Lenstra. Factoring multivariate integral polynomials. *Theoretical Computer Science*, vol. 34, pp. 207-213, 1984.

[3] M. Sudan. Decoding of Reed Solomon Codes beyond the Error-Correction Bound. *Journal of Complexity* , vol. 13(1), pp. 180-193, 1997.