Encoding and Decoding Reed-Solomon Codes

John McCall

February 28, 2014

1 Introduction

Let F be a field. Choose nonzero elements $v_1, ..., v_n \in F$ and distinct elements $\alpha_1, ..., \alpha_n \in F$. Set $\mathbf{v} = (v_1, ..., v_n)$ and $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_n)$ For $0 \leq k \leq n$ *generalized Reed-Solomon codes* are defined as

$$\mathrm{GRS}_{n,k}(\boldsymbol{\alpha}, \mathbf{v}) = \{(v_1 f(\alpha_1), v_2 f(\alpha_2), ..., v_n f(\alpha_n)) \mid f(x) \in F[x]_k\}.$$

Here $F[x]_k$ is the set of polynomials in $F[x]$ with degree less than k. If $f(x)$ is a polynomial, then $\mathbf{f}$ is its associated codeword. This codeword is also dependent on $\boldsymbol{\alpha}$and$\mathbf{v}$. We can write

$$\mathbf{ev}_{\boldsymbol{\alpha},\mathbf{v}}(f(x)) = (v_1 f(\alpha_1), v_2 f(\alpha_2), ..., v_n f(\alpha_n)),$$

where $f = \mathbf{ev}_{\boldsymbol{\alpha},\mathbf{v}}(f(x))$ when the polynomial $f(x)$ is evaluated at $\boldsymbol{\alpha}$ and scaled by $\mathbf{v}$.

The distance between two codewords is defined as the number of symbols in which the sequences differ, in other words it is the Hamming distance. The minimum distance between two codewords for GRS codes is

$$\mathrm{d}_{\min} = n - k + 1.$$

A key concept is that any codeword which has up to k entries equal to 0 corresponds to a polynomial of degree less than k whose values matching the 0 polynomial in k points must be the 0 polynomial. This is true since any polynomial of degree less than k is uniquely determined by its values at k distinct points. Which means that for any n-tuple $\mathbf{f}$, we can reconstruct the polynomial $f(x)$ of degree less than n such that $\mathbf{f} = \mathbf{ev}_{\boldsymbol{\alpha},\mathbf{v}}(f(x))$. Let

$$L(x) = \prod_{i=1}^{n} (x - \alpha_i)$$

and

$$L_i(x) = L(x)/(x - \alpha_i) = \prod_{j \neq i} (x - \alpha_j).$$

Both $L(x)$ and $L_i(x)$ are *monic* polynomials of degrees n and $n-1$, respectively. A polynomial is monic if the leading coefficient is equal to 1. The vector $\mathbf{f}$ has i^{th} coordinate $v_i f(\alpha_i)$. This is enough information to use Lagrange interpolation [insert reference to Lagrange interpolation here] to calculate

$$f(x) = \sum_{i=1}^{n} \frac{L_i(x)}{L_i(\alpha_i)} f(\alpha_i).$$

2 Definitions

Before we delve into encoding and decoding a Reed-Solomon Code, there are many terms that we need to define.

3 Encoding

4 Decoding