Reed-Solomon Decoding for Non-Experts

Sarah Helmbrecht

May 2023

Abstract

Reed-Solomon codes are an important topic in many introductory coding theory courses. While defining them and explaining their applications are approachable topics for computer science students, decoding algorithms tend to be too complicated to teach in university courses. In this paper, we discuss a decoding algorithm introduced by Maria Bras-Amorós that is intended to be appropriate for college students.

1 Introduction

Transmission and storage can introduce distortions and errors into data. Error control codes are used to detect and correct these errors. Error control codes work by sending some redundancy along with the original data. This redundancy does increase the cost of data transmission, and the aim of coding theory is to create codes that maintain a low transmission cost while having good correction capacity. Coding theory also aims to design algorithms that allow receivers to decode the original data.

Reed-Solomon codes are a powerful method for preventing corruption of data. Better techniques have since been developed, but Reed-Solomon codes are still the most universal error control codes. Most new techniques are heavily based on Reed-Solomon codes.

In her paper "A Decoding Approach to Reed-Solomon Codes from Their Definition," Maria Bras-Amorós introduces a decoding approach for Reed-Solomon codes that is intended to be digestible to non-experts **[1].** In this paper, we will walk through the bulk of the content of Bras-Amorós's paper, adding additional explanations and commentary with the intention of making Reed-Solomon decoding even more approachable.

Section 2 will provide some background on the mathematics that are required to understand Reed-Solomon codes and the decoding algorithm presented by Bras-Amorós. This includes basic information about fields, vectors and vector spaces, matrices, and polynomials. Section 3 will move into background information that is more specific to coding theory, mirroring Bras-Amorós's section entitled "Some Background on Coding Theory." In Section 4, we will define Reed-Solomon codes from four different perspectives and provide relevant theorems and lemmas. Section 5 will establish the new decoding approach on which Bras-Amorós's central thesis rests. We will omit the proofs of many of the theorems and lemmas that are stated, focusing instead on applications and examples. The definitions, lemmas, and examples in Sections 3, 4, and 5 will be primarily drawn from Bras-Amorós's paper, although additional explanations, comments, and steps will be added.

2 Background

A **field** is a set endowed with operations $+$ and $\cdot$, which satisfy the properties of additive and multiplicative associativity, commutativity, identities, and inverses. A number $q \in \mathbb{Z}$ is **prime** if and only if $q > 1$ and the only positive integers that divide q are q and 1. Applying a **modulus** mod$q$ to a number means to find the number $0 \leq \text{mod}q \leq q-1$ such that mod$q$ is the remainder of the number divided by q **[3].**

A **vector space** is a subspace of the field $\mathbb{R}^n$ that satisfies the properties of additive commutativity, additive associativity, zero element, additive inverse, distributivity, and multiplicative identity. An element of a vector space is called a **vector**, a quantity with both magnitude and direction **[2].**

The **dot product** of two vectors $\vec{u} = (u_0, u_1, \ldots, u_{n-1}), \vec{v} = (v_0, v_1, \ldots, v_{n-1}) \in \mathbb{F}_q^n$ is the scalar $\vec{u} \cdot \vec{v} = u_0v_0 + u_1v_1 + \ldots + u_{n-1}v_{n-1} \in \mathbb{F}_q$.

A **matrix** is a rectangular array of numbers. A $k \times n$ matrix has k rows and n columns. A matrix can be used to represent a linear system of equations, where each row represents an equation and each column represents a variable. Matrices can be manipulated using the three Elementary Row operations, which include multiplying a row by a nonzero constant, switching any two rows, and adding a constant multiple of one row to another row.

To solve a matrix A, we can get it into **echelon form**, where every row of A that consists entirely of zeroes lies at the bottom of the matrix, and in each row of A that contains a nonzero element, the first nonzero element (leading entry) lies to the right of the leading entry of the preceding row. We can get A into echelon form using **Gaussian elimination:**

1. Locate the first column of A that has a nonzero element.
2. If the first entry in this column is 0, then switch the first row of A with a row where the corresponding entry is nonzero.
3. Replace the entries below this nonzero entry with zeroes by adding multiples of the first row to lower rows.
4. Perform steps 1-3 on lower right matrix A_1.
5. Repeat this cycle of steps until an echelon matrix is obtained.

The **determinant** of a $k \times k$ matrix $A = [a_{ij}]$ is $\det A = a_{11}A_{11} + a_{12}A_{12} + \ldots + a_{1k}A_{1k}$.

If and only if the determinant of a square matrix A is nonzero, A has an inverse A^{-1} such that $A \times A^{-1} = A^{-1} \times A = I$, where I is the identity matrix with ones along the diagonal and zeroes elsewhere.

A **polynomial** of degree k is a function of the form $P(x) = a_0 + a_1x + a_2x^2 + \ldots + a_kx^k$.

To add or subtract two polynomials, you add or subtract the coefficients of like terms. For example, $(2x^2 + 3x + 2) + (3x^2 - 5x - 1) = (2 + 3)x^2 + (3 + (-5))x + (2 + (-1)) = 5x^2 - 2x + 1$ **[6].**

To multiply two polynomials, you multiply every term of one polynomial with every term of the other, then add the results. For example, $(2x + 3y) \times (4x - 5y) = 2x(4x - 5y) + 3y(4x - 5y) = 2x(4x) + 2x(-5y) + 3y(4x) + 3y(-5y) = 8x^2 - 10xy + 12xy - 15y^2 = 8x^2 + 2xy - 15y^2$.

To divide two polynomials, you can use long division. For example, to solve $(6x^2 + 10x - 24)/(2x + 6)$: **[4].**

$$
\begin{array}{r|l}
2x+6 & 3x - 4 \\
\hline
 & 6x^2 + 10x - 24 \\
 & -6x^2 - 18x \\
\hline
 & -8x - 24 \\
 & 8x + 24 \\
\hline
 & 0
\end{array}
$$

Any polynomial function of degree $k - 1$ is uniquely defined by any k points that lay on the function. For example, the polynomial $f(x) = 2 + 3x - 5x^2 + x^3$ includes the 4 points $(-1, 7), (0, 2), (1, 1)$, and $(2, -4)$. This gives the following set of 4 linear equations:

$a_0 + a_1(-1) + a_2(-1)^2 + a_3(-1)^3 = -7$
$a_0 + a_1(0) + a_2(0)^2 + a_3(0)^3 = 2$
$a_0 + a_1(1) + a_2(1)^2 + a_3(1)^3 = 1$
$a_0 + a_1(2) + a_2(2)^2 + a_3(2)^3 = -4$
This can be converted into a matrix, and we can use Gaussian elimination to solve it:

$$\left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & -7 \\ 1 & 0 & 0 & 0 & 2 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & -4 \end{array}\right) = \left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & -7 \\ 0 & 1 & -1 & 1 & 9 \\ 0 & 2 & 0 & 2 & 8 \\ 0 & 3 & 3 & 9 & 3 \end{array}\right) = \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & -1 & 1 & 9 \\ 0 & 0 & 2 & 0 & -10 \\ 0 & 0 & 6 & 6 & -24 \end{array}\right)$$

$$= \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & -1 & 1 & 9 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 6 & 6 & -24 \end{array}\right) = \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & -1 & 1 & 9 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 6 & 6 \end{array}\right) = \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 1 & 4 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right)$$

$$= \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right)$$

This yields $a_0 = 2$, $a_1 = 3$, $a_2 = -5$, and $a3 = 1$. This gives our original polynomial $2 + 3x - 5x^2 + x^3$. This shows that the 4 points uniquely define a polynomial of degree 3 **[7].**

A **monic** polynomial is a polynomial whose term with the highest power has a coefficient of 1 **[5].**

3 Coding Theory

The **transmission alphabet** $\mathbb{F}_q$ is a finite field of symbols. These symbols correspond to the data that is sent and received during data transmission. A **prime field** $\mathbb{F}_q$ is identified by the ring $\{0, 1, \ldots, q-1\}$, equipped with $+$ and $\cdot$ modulo q. If $\mathbb{F}_q$ is a prime field, then there exists a **primitive element** $\alpha \in \mathbb{F}_q$ such that all of the powers of α with exponent smaller than $q-1$ are different. We can rewrite the prime field $\mathbb{F}_q$ in terms of the primitive element: $\mathbb{F}_q = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^{q-2}\}$.

A **linear code** C of **length** n over $\mathbb{F}_q$ is a vector subspace of $\mathbb{F}_q^n$. Each vector $c \in C$ is called a **code word.** The **dimension** k of C is the dimension of the subspace $C \subseteq \mathbb{F}_q^n$. The linear code C contains q^k code words c.

To encode a word $c \in C$ of k symbols from $\mathbb{F}_q$, we multiply c by the **generator matrix** G of the linear code C. G is a $k \times n$ matrix whose k rows are a set of vectors that generate the linear code C. The generator matrix is not unique, so a generator matrix that has been transformed using elementary row operations still generates C. All code words $c \in C$ are linear combinations of the rows of G.

The **dual code** of C is the vector subspace $C^\perp = \{v \in \mathbb{F}_q^n | v \cdot c = 0, \forall c \in C\}$. $C^\perp$ is a linear code with length n (same as C) and dimension $n - k$. A matrix H that generates the dual code $C^\perp$ is called a **parity-check matrix** of C. We can rewrite the linear code C as $C = \{c \in \mathbb{F}_q^n | c \cdot h = 0, \forall \text{ rows } h \in H\}$.

The **Hamming distance** between two code words of the same length n is the number of positions in which their symbols differ. Given an input vector u of the same length n as the code, decoding algorithms aim to output a code word $c \in C$ while minimizing the Hamming distance between u and c. The **weight** of a word is its Hamming distance from $\vec{0}$. This represents the number of nonzero symbols that the word contains.

The **minimum distance** d of a linear code C can be defined in three ways:

1. The minimum Hamming distance between two words of C.
2. The minimum weight of nonzero words of C.

3. The minimum number of linearly dependent columns of the parity-check matrix H of C.

The minimum distance a code represents its capacity to correct errors. If at most $\lfloor \frac{d-1}{2} \rfloor$ errors are added to a code word $c \in C$, then $\lfloor \frac{d-1}{2} \rfloor$ errors can be corrected.

The **Singleton bound** states that for a linear code with length n, minimum distance d, and dimension k, we have $k \leq n - d + 1$. **Maximum distance separable (MDS) codes** are those that attain equality, so $k = n - d + 1$.

Our final background definition is of Vandermonde matrices. Given primitive elements $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{F}_q$, the **Vandermonde matrix** of $\alpha_1, \alpha_2, \ldots, \alpha_n$ of order r is defined as:

$$V_r(\alpha_1, \alpha_2, \ldots, \alpha_n) = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{r-1} & \alpha_2^{r-1} & \ldots & \alpha_n^{r-1} \end{pmatrix}.$$

The determinant of $V_n(\alpha_1, \alpha_2, \ldots, \alpha_n)$ satisfies $|V(\alpha_1, \alpha_2, \ldots, \alpha_n)| = \prod_{1 \leq i < j \leq n} (\alpha_j - \alpha_i)$. Therefore, $V_n(\alpha_1, \alpha_2, \ldots, \alpha_n)$ has an inverse matrix if and only if $\alpha_i \neq \alpha_j, \forall 1 \leq i < j \leq n$.

Consider the alphabet $\mathbb{F}_7 = 0, 1, 2, 3, 4, 5, 6$. This set has addition and multiplication modulo 7.

Since 7 is a prime number, $\mathbb{F}_q$ is a prime field. This means that it must have a primitive element. $\alpha = 5$ is a primitive element of $\mathbb{F}_q$ because $5^0 \text{mod} 7 = 1$, $5^1 \text{mod} 7 = 5$, $5^2 \text{mod} 7 = 4$, $5^3 \text{mod} 7 = 6$, $5^4 \text{mod} 7 = 2$, and $5^5 \text{mod} 7 = 3$, all of which are different.

The matrix $G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix}$ is a generator matrix of a code $C \in \mathbb{F}_7$. C has length $n = 6$ and dimension $k = 2$. Suppose that the code word we want to encode is $c = 110256 \in C$. We split C into blocks of $k = 2$, and multiply each block by G :

$$\begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 2 & 6 & 5 & 0 & 3 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 2 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 2 & 3 & 1 & 5 & 4 & 6 \end{pmatrix}$$

$$\begin{pmatrix} 5 & 6 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 4 & 0 & 1 & 6 & 3 & 2 \end{pmatrix}$$

After encoding, C will be represented by the code 265034231546401632.

Consider code words 265034 and 231546. Both begin with the digit 2, but have different digits in all 5 other positions. Therefore, their Hamming distance is 5. Now consider code words 111111 and 265034. They have different digits in all 6 positions, so their Hamming distance is 6. Any code word $c \in C$ is a linear combination of the two rows of G : $c = x_1 \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} + x_2 \begin{pmatrix} 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix}$. So any vector $c \in C$ is either constant or all its elements are different, meaning that any two words in C have a Hamming distance of either 5 or 6. This means that the minimum distance of C is 5.

Now, we can look at how Reed-Solomon encoding works. We have a transmission alphabet $\mathbb{F}_q^n$, and we want to transmit k of its nonzero elements. We find a polynomial of degree less than k that takes the values of the elements that we are transmitting when evaluated at k. Then, we add the redundancy to the original elements. This redundancy consists of the evaluation of the polynomial at the remaining $q - 1 - k$ nonzero values of $\mathbb{F}_q^n$.

4 Defining Reed-Solomon Codes

We will define Reed-Solomon codes in four different ways in order to provide a basis for proving a new decoding approach. We will also provide relevant theorems and lemmas to support these definitions.

Definition 4.1. *Let q be a prime power, meaning that q is a power of a prime number. Let $\mathbb{F}_q$ be the field with q elements. Let α be a primitive element of $\mathbb{F}_q$. Then $\mathbb{F}_q = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^{q-2}\}$. Let $n = q - 1$ be the length of code word C. The Reed-Solomon code over $\mathbb{F}_q$ of dimension k, $RS_{q,\alpha}(k)$, is the linear code of $\mathbb{F}_q^n$ with generator matrix:*

$$G = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \alpha & \alpha^2 & \ldots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \ldots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \alpha^{(k-1)2} & \ldots & \alpha^{(k-1)(n-1)} \end{pmatrix}.$$

For example, we have already shown that $\alpha = 5$ is a primitive element of $\mathbb{F}_7$, the prime field with $q = 7$. Then $\mathbb{F}_7 = \{0, 1, 5, 5^2, 5^3, 5^4, 5^5\} = \{0, 1, 5, 4, 6, 2, 3\}$. Let $n = 7 - 1 = 6$ be the length of code word C. Then the Reed-Solomon code $RS_{7,5}(2)$ over $\mathbb{F}_7$ of dimension $k = 2$ is the code C with generator matrix $G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 5^2 & 5^3 & 5^4 & 5^5 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix}$.

Definition 4.2. *The Reed-Solomon code over $\mathbb{F}_q$ of dimension k, $RS_{q,\alpha}(k)$, is the linear code of $\mathbb{F}_q^n$ with parity check matrix H defined as follows:*

$$H = \begin{pmatrix} 1 & \alpha & \alpha^2 & \ldots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \ldots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{n-k} & \alpha^{(n-k)2} & \ldots & \alpha^{(n-k)(n-1)} \end{pmatrix}.$$

For example, the Reed-Solomon code over $\mathbb{F}_7$ of dimension $k = 2$, $RS_{7,5}(2)$, is the code C with parity-check matrix defined using the primitive element $\alpha = 5$:

$$H = \begin{pmatrix} 1 & 5 & 5^2 & 5^3 & 5^4 & 5^5 \\ 1 & 5^2 & 5^4 & 5^6 & 5^8 & 5^{10} \\ 1 & 5^3 & 5^6 & 5^9 & 5^{12} & 5^{15} \\ 1 & 5^4 & 5^8 & 5^{12} & 5^{16} & 5^{20} \end{pmatrix} = \begin{pmatrix} 1 & 5 & 4 & 6 & 2 & 3 \\ 1 & 4 & 2 & 1 & 4 & 2 \\ 1 & 6 & 1 & 6 & 1 & 6 \\ 1 & 2 & 4 & 1 & 2 & 4 \end{pmatrix}.$$

Lemma 4.3. *The minimum distance of $RS_{q,\alpha}(k)$ is $n - k + 1$. Therefore, $RS_{q,\alpha}(k)$ is an MDS code.*

For example, we have already stated that the minimum distance of $RS_{7,5}(2)$ is 5. Since $n = 6$ and $k = 2$, the minimum distance of $RS_{7,5}(2)$ is equal to $n - k + 1 = 6 - 2 + 1 = 5$.

Definition 4.4. *Consider the set $\mathbb{F}_q[x]^{<k}$ of polynomials with coefficients in $\mathbb{F}_q$ and of degree less than k. An element $a \in \mathbb{F}_q[x]^{<k}$ has the form $a = a_0 + a_1 x + \ldots a_{k-1} x^{k-1}$, with $a_i \in \mathbb{F}_q$. The Reed-Solomon code over $\mathbb{F}_q$ and of dimension k, $RS_{q,\alpha}(k)$, is the set $\{(a(1), a(\alpha), a(\alpha^2), \ldots, a(\alpha^{n-1}))|a \in \mathbb{F}_q[x]^{<k}\}$.*

For example, we have already shown that the code 110256 is encoded as 265034, 231546, and 401632 using the generator matrix $G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 4 & 6 & 2 & 3 \end{pmatrix}$. 265034 is the encoding of the block 11. 11 corresponds to the polynomial $1 + x$. If we evaluate $1 + x$ at $5^0, 5^1, 5^2, 5^3, 5^4$, and 5^5, which are equal to $1, 5, 4, 6, 2$, and 3, then we get the following:
At $x = 5^0 = 1, 1 + x = 1 + 1 = 2$.
At $x = 5^1 = 5, 1 + x = 1 + 5 = 6$.
At $x = 5^2 = 4, 1 + x = 1 + 4 = 5$.
At $x = 5^3 = 6, 1 + x = 1 + 6 = 0$.
At $x = 5^4 = 2, 1 + x = 1 + 2 = 3$.

At $x = 5^5 = 3, 1 + x = 1 + 3 = 4$.
This gives us the same code, 265034, that we computed using the generator matrix.
The same can be done for encoding the block 02 to get the code 231546. 02 corresponds to the interpolation polynomial $0 + 2x = 2x$.
At $x = 5^0 = 1, 2x = 2(1) = 2$.
At $x = 5^1 = 5, 2x = 2(5) = 3$.
At $x = 5^2 = 4, 2x = 2(4) = 1$.
At $x = 5^3 = 6, 2x = 2(6) = 5$.
At $x = 5^4 = 2, 2x = 2(2) = 4$.
At $x = 5^5 = 3, 2x = 2(3) = 6$.
Again, we can do the same to get 401632 from the block 56. 56 corresponds to the interpolation polynomial $5 + 6x$.
At $x = 5^0 = 1, 5 + 6(1) = 5 + 6 = 4$.
At $x = 5^1 = 5, 5 + 6(5) = 5 + 2 = 0$.
At $x = 5^2 = 4, 5 + 6(4) = 5 + 3 = 1$.
At $x = 5^3 = 6, 5 + 6(6) = 5 + 1 = 6$.
At $x = 5^4 = 2, 5 + 6(2) = 5 + 5 = 3$.
At $x = 5^5 = 3, 5 + 6(3) = 5 + 4 = 2$.
We can use this process to pass the elements of $\mathbb{F}_q$ to an interpolation polynomial to get the Reed-Solomon encoding of a code word.

Next, we need to know how to reliably find the interpolation polynomial for a vector. For each vector $u = (u_0, u_1, \ldots, u_{n-1}) \in \mathbb{F}_q^n$, there is a polynomial f_u of degree at most $n-1$ such that $f_u(\alpha^i) = u_i, \forall i \in \{0, 1, \ldots, n-1\}$. Let f_i be the interpolation polynomial of the $i-$th standard basis vector. Then $f_i = \prod_{j=0, j\neq i}^{n-1} \frac{x-\alpha^j}{\alpha^i - \alpha^j}$. Then we can compute $f_u = \sum_{i=1}^{n-1} u_i f_i$. The polynomial f_u is unique because if $f_u = a_0 + a_1 x + \ldots + a_{n-1}x^{n-1}$, then $a_0, a_1, \ldots, a_{n-1}$ are the solution of the linear system:

$$\begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \alpha & \alpha^2 & \ldots & \alpha^{(n-1)} \\ 1 & \alpha^2 & \alpha^4 & \ldots & \alpha^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \\ 1 & \alpha^{n-1} & \alpha^{2(n-1)} & \ldots & \alpha^{(n-1)(n-1)} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{pmatrix}.$$

The matrix of this system has a Vandermonde structure and is invertible. This means that for any vector $u \in \mathbb{F}_q^n$, there exists some unique $f \in \mathbb{F}_q[x]$ of degree less than n such that $u = (f(1), f(\alpha), f(\alpha^2), \ldots, f(\alpha^{n-1}))$.

For example, in the field $\mathbb{F}_7$, we can use the formula $f_i = \prod_{j=0, j\neq i}^{n-1} \frac{x-\alpha^j}{\alpha^i - \alpha^j}$ with $\alpha = 5$ to get the following:
$f_0 = \prod_{j=1}^{j=5} \frac{x-5^j}{1-5^j} = 6x^5 + 6x^4 + 6x^3 + 6x^2 + 6x + 6$,
$f_1 = \prod_{j=0, j\neq 1}^{j=5} \frac{x-5^j}{5-5^j} = 2x^5 + 3x^4 + x^3 + 5x^2 + 4x + 6$,
$f_2 = \prod_{j=0, j\neq 2}^{j=5} \frac{x-5^j}{4-5^j} = 3x^5 + 5x^4 + 6x^3 + 3x^2 + 5x + 6$,
$f_3 = \prod_{j=0, j\neq 3}^{j=5} \frac{x-5^j}{6-5^j} = x^5 + 6x^4 + x^3 + 6x^2 + x + 6$,
$f_4 = \prod_{j=0, j\neq 4}^{j=5} \frac{x-5^j}{2-5^j} = 5x^5 + 3x^4 + 6x^3 + 5x^2 + 3x + 6$,
$f_5 = \prod_{j=0, j\neq 5}^{j=5} \frac{x-5^j}{3-5^j} = 4x^5 + 5x^4 + x^3 + 3x^2 + 2x + 6$.

So for any vector $u \in \mathbb{F}_7^6$, we can compute the coefficients of f_u as the product of u times the following matrix:

$$\begin{pmatrix} 6 & 6 & 6 & 6 & 6 & 6 \\ 6 & 4 & 5 & 1 & 3 & 2 \\ 6 & 5 & 3 & 6 & 5 & 3 \\ 6 & 1 & 6 & 1 & 6 & 1 \\ 6 & 3 & 5 & 6 & 3 & 5 \\ 6 & 2 & 3 & 1 & 5 & 4 \end{pmatrix}.$$

So the coefficients of the polynomial interpolating $u = (4, 2, 1, 6, 3, 2)$ can be found as follows:

$$\begin{pmatrix} 6 & 6 & 6 & 6 & 6 & 6 \\ 6 & 4 & 5 & 1 & 3 & 2 \\ 6 & 5 & 3 & 6 & 5 & 3 \\ 6 & 1 & 6 & 1 & 6 & 1 \\ 6 & 3 & 5 & 6 & 3 & 5 \\ 6 & 2 & 3 & 1 & 5 & 4 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 1 \\ 6 \\ 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \\ 3 \\ 2 \\ 6 \\ 4 \end{pmatrix}.$$

This means that the coefficients of the interpolation polynomial, in increasing order, are $(3, 0, 3, 2, 6, 4)$. The interpolation polynomial is therefore $3 + 3x^2 + 2x^3 + 6x^4 + 4x^5$.

Code word checking: a vector $\vec{u} = (u_0, u_1, \ldots, u_{n-1}) \in \mathbb{F}_q^n$ is a code word if and only if the degree of its interpolation polynomial f_u is less than k.

For example, if we want to determine if 265034 is a code word of $RS_{7,5}(2)$, we must first find its interpolation polynomial. We do this the same way we did above:

$$\begin{pmatrix} 6 & 6 & 6 & 6 & 6 & 6 \\ 6 & 4 & 5 & 1 & 3 & 2 \\ 6 & 5 & 3 & 6 & 5 & 3 \\ 6 & 1 & 6 & 1 & 6 & 1 \\ 6 & 3 & 5 & 6 & 3 & 5 \\ 6 & 2 & 3 & 1 & 5 & 4 \end{pmatrix} \begin{pmatrix} 2 \\ 6 \\ 5 \\ 0 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Therefore, 265034 has an interpolation polynomial of $x + 1$. This has a degree of less than $k = 2$, so 265034 is a code word of $RS_{7,5}(2)$.

Now consider the word 025606. We find its interpolation polynomial the same way:

$$\begin{pmatrix} 6 & 6 & 6 & 6 & 6 & 6 \\ 6 & 4 & 5 & 1 & 3 & 2 \\ 6 & 5 & 3 & 6 & 5 & 3 \\ 6 & 1 & 6 & 1 & 6 & 1 \\ 6 & 3 & 5 & 6 & 3 & 5 \\ 6 & 2 & 3 & 1 & 5 & 4 \end{pmatrix} \begin{pmatrix} 0 \\ 2 \\ 5 \\ 6 \\ 0 \\ 6 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 6 \\ 0 \end{pmatrix}.$$

Therefore, 025606 has an interpolation polynomial of $6x^4 + 2x^3 + 2x^2 + 2x + 2$. This has degree of greater than $k = 2$, so 025606 is not a code word of $RS_{7,5}(2)$.

Definition 4.5. *Consider the set $\mathbb{F}_q[x]^{<n}$ of all polynomials with coefficients in $\mathbb{F}_q$ of degree less than n. An element $u \in \mathbb{F}_q[x]^{<n}$ is of the form $u = u_0 + u_1x + \ldots + u_{n-1}x^{n-1}$ with $u_i \in \mathbb{F}_q$. The Reed-Solomon code over $\mathbb{F}_q$ of dimension k, $RS_{q,\alpha}(k)$, is the set of vectors $u = (u_0, \ldots, u_{n-1})$ in $\mathbb{F}_q^n$ s.t. the polynomial $u_0 + u_1x + \ldots + u_{n-1}x^{n-1}$ vanishes at $\alpha^j, \forall j$ with $1 \leq j \leq n - k$.*

Code word checking: a vector $\vec{u} = (u_0, u_1, \ldots, u_{n-1}) \in \mathbb{F}_q^n$ is a code word if and only if $u(\alpha^i) = 0$ for all i such that $1 \leq i \leq n - k$.

For example, in order to check whether the word 342650 is a code word of $RS_{7,5}(2)$, we evaluate its associated polynomial $u(x) = 3 + 4x + 2x^2 + 6x^3 + 5x^4$ at $5, 5^2, 5^3$, and 5^4 :

$u(5^1) = u(5) = 3 + 4(5) + 2(5^2) + 6(5^3) + 5(5^4) = 3 + 4(5) + 2(4) + 6(6) + 5(2) = 3 + 6 + 1 + 1 + 3 = 0.$
$u(5^2) = u(4) = 3 + 4(4) + 2(4^2) + 6(4^3) + 5(4^4) = 3 + 4(4) + 2(2) + 6(1) + 5(4) = 3 + 2 + 4 + 6 + 6 = 0.$
$u(5^3) = u(6) = 3 + 4(6) + 2(6^2) + 6(6^3) + 5(6^4) = 3 + 4(6) + 2(1) + 6(6) + 5(1) = 3 + 3 + 2 + 1 + 5 = 0.$
$u(5^4) = u(2) = 3 + 4(2) + 2(2^2) + 6(2^3) + 5(2^4) = 3 + 4(2) + 2(4) + 6(1) + 5(2) = 3 + 1 + 1 + 6 + 3 = 0.$

Therefore, 342650 is a code word of $RS_{7,5}(2)$.

Lemma 4.6. *Suppose that α is a primitive element of a finite field of q elements and let $n = q - 1$. The polynomials $f_i = \prod_{j=0, j\neq i}^{n-1} \frac{x-\alpha^j}{\alpha^i - \alpha^j}$ satisfy $f_i = -(\alpha^i x^{n-1} + \alpha^{2i} x^{n-2} + \alpha^{3i} x^{n-3} + \ldots + \alpha^{(n-1)i} x + \alpha^{ni})$.*

Lemma 4.7. *The inverse of the map from $\mathbb{F}^n$ to $\mathbb{F}^n$ defined by $(v_0, \ldots, v_{n-1}) \mapsto (v(\alpha^0), v(\alpha), v(\alpha^2), \ldots, v(\alpha^{n-1}))$ is the map $(u_0, \ldots, u_{n-1}) \mapsto (-u(\alpha^n), -u(\alpha^{n-1}), -u(\alpha^{n-2}), \ldots, -u(\alpha))$, where $v(\beta)$ is the evaluation of $v_0 + v_1x + \ldots + v_{n-1}x^{n-1}$ at β and $u(\beta)$ is the evaluation of $u_0 + u_1x + \ldots + u_{n-1}x^{n-1}$ at β.*

As an example of Lemma 4.7, the word $(u_0, u_1, u_2, u_3, u_4, u_5) = (5, 4, 0, 1, 2, 0)$ has polynomial $u = 5 + 4x + x^3 + 2x^4$. Evaluating the polynomial at $5^1, 5^2, 5^3, 5^4, 5^5$, and 5^6, we get the following:
$u(5^1) = u(5) = 5 + 4(5) + 5^3 + 2(5^4) = 5 + 4(5) + 6 + 2(2) = 5 + 6 + 6 + 4 = 0$,
$u(5^2) = u(4) = 5 + 4(4) + 4^3 + 2(4^4) = 5 + 4(4) + 1 + 2(4) = 5 + 2 + 1 + 1 = 2$,
$u(5^3) = u(6) = 5 + 4(6) + 6^3 + 2(6^4) = 5 + 4(6) + 6 + 2(1) = 5 + 3 + 6 + 2 = 2$,
$u(5^4) = u(2) = 5 + 4(2) + 2^3 + 2(2^4) = 5 + 4(2) + 1 + 2(2) = 5 + 1 + 1 + 4 = 4$,
$u(5^5) = u(3) = 5 + 4(3) + 3^3 + 2(3^4) = 5 + 4(3) + 6 + 2(4) = 5 + 5 + 6 + 1 = 3$,
$u(5^6) = u(1) = 5 + 4(1) + 1^3 + 2(1^4) = 5 + 4(1) + 1 + 2(1) = 5 + 4 + 1 + 2 = 5$.
By Lemma 4.7, it should be the case that if $v(x) = -5 - 3x - 4x^2 - 2x^3 - 2x^4 = 2 + 4x + 3x^2 + 5x^3 + 5x^4$, then $u = (v(1), v(5), v(5^2), v(5^3), v(5^4), v(5^5))$. To verify this, we show that $(v(1), v(5), v(5^2), v(5^3), v(5^4), v(5^5)) = (5, 4, 0, 1, 2, 0)$:
$v(5^0) = v(1) = 2 + 4(1) + 3(1^2) + 5(1^3) + 5(1^4) = 2 + 4(1) + 3(1) + 5(1) + 5(1) = 2 + 4 + 3 + 5 + 5 = 5$,
$v(5^1) = v(5) = 2 + 4(5) + 3(5^2) + 5(5^3) + 5(5^4) = 2 + 4(5) + 3(4) + 5(6) + 5(2) = 2 + 6 + 5 + 2 + 3 = 4$,
$v(5^2) = v(4) = 2 + 4(4) + 3(4^2) + 5(4^3) + 5(4^4) = 2 + 4(4) + 3(2) + 5(1) + 5(4) = 2 + 2 + 6 + 5 + 6 = 0$,
$v(5^3) = v(6) = 2 + 4(6) + 3(6^2) + 5(6^3) + 5(6^4) = 2 + 4(6) + 3(1) + 5(6) + 5(1) = 2 + 3 + 3 + 2 + 5 = 1$,
$v(5^4) = v(2) = 2 + 4(2) + 3(2^2) + 5(2^3) + 5(2^4) = 2 + 4(2) + 3(4) + 5(1) + 5(2) = 2 + 1 + 5 + 5 + 3 = 2$,
$v(5^5) = v(3) = 2 + 4(3) + 3(3^2) + 5(3^3) + 5(3^4) = 2 + 4(3) + 3(2) + 5(6) + 5(4) = 2 + 5 + 6 + 2 + 6 = 0$.
This demonstrates that Lemma 4.7 holds for the word $(u_0, u_1, u_2, u_3, u_4, u_5) = (5, 4, 0, 1, 2, 0)$.

Lemmas 4.6 and 4.7 are important because they establish the connection between the coefficients of an interpolation polynomial over a finite field and the evaluation of the polynomial at all the nonzero elements of the finite field.

5 A New Decoding Approach

Now that we have defined Reed-Solomon codes in several ways, we have the tools to approach the decoding of words. We will discuss decoding using interpolation polynomials, as established in Definition 4.4. Our proofs, however, will use Definition 4.5, focusing on polynomial evaluation.

First, let $\mathbb{F}_q[x]^{<d}$ be the set of all polynomials with coefficients in the alphabet $\mathbb{F}_q$ and degree strictly less than d. Let $\mathbb{F}_q[x]^{\leq d}_{\geq d'}$ be the set of all polynomials with coefficients in the alphabet $\mathbb{F}_q$ of degree less than d and greater than or equal to d'.

Consider receiving the message $u \in \mathbb{F}_q^n$ with interpolation polynomial f_u. In order to decode u into a word c, we want to find $c \in RS_{q,\alpha}(k)$ such that the Hamming distance between u and c is minimized. We know that all words $c \in RS_{q,\alpha}(k)$ are the evaluation of polynomials of degree less than k at the nonzero elements of $\mathbb{F}_q$. Therefore, we can decode u, by finding the interpolation polynomial of c. This will be a polynomial $g_c \in \mathbb{F}_q[x]^{<k}$ such that $f_u - g_c$ has a maximum number of nonzero roots.

We can split the terms of f_u into those that have degree less than k and those that have degree greater than or equal to k. Let $f_u = h_u + g_u$ for unique polynomials h_u and g_u. Let $h_u \in \mathbb{F}_q[x]^{\leq n}_{\geq k}$, meaning that h_u is made up of the terms with degree greater than or equal to k and less than n. Let $g_u \in \mathbb{F}_q[x]^{<k}$, meaning that g_u is made up of the terms with degree less than k. Consider a polynomial g_{h_u} that maximizes the number of nonzero roots of $f_u = h_u + g_u$. Then, for any $g' \in \mathbb{F}_q[x]^{<k}$, the number of nonzero roots of $h_u + g_{h_u}$ is greater than or equal to the number of nonzero roots of $h_u + g'$. Then, the interpolation polynomial g_c of c satisfies the equation $g_c = g_u - g_{h_u}$.

Let e be the minimum weight word such that $u - e \in RS_{q,\alpha}(k)$. Then $e = u - c$, and the interpolation polynomial of e is $f_e = f_u - g_c = h_u + g_{h_u}$.

Now, we define the set of polynomials $\Lambda = \{\lambda \in \mathbb{F}_q[x] : \lambda(h_u + g)$ vanishes at all $\mathbb{F}_q \backslash \{0\}$, for some $g \in \mathbb{F}_q[x]^k\}$. Since we know that $x^n - 1 = \prod_{\gamma \in \mathbb{F}_q \backslash \{0\}} (x - \gamma)$, we can also rewrite this as $\Lambda = \{\lambda \in \mathbb{F}_q[x] : (x^n - 1)$ divides $\lambda(h_u + g)$ for some $g \in \mathbb{F}_q[x]^{<k}\}$. Since $x^n - 1$ belongs to Λ, we know that Λ is a nonempty set.

Recall that a monic polynomial is one whose term with the highest power has a coefficient of 1.

Theorem 5.1. *Let λ_u be a monic polynomial with minimum degree among the polynomials in Λ. For a polynomial $g \in \mathbb{F}_q[x]^{<k}$, if $(x^n - 1)$ divides $\lambda_u(h_u + g)$, then the number of nonzero roots of $h_u + g$ is greater than or equal to the number of nonzero roots of $h_u + g'$ for any $g' \in \mathbb{F}_q[x]^{<k}$.*

Lemma 5.2. *Let λ_u be a monic polynomial with minimum degree among the polynomials in Λ. The non-leading coefficients of λ_u give a solution to the following linear system:*

$$\begin{pmatrix} u(\alpha) & u(\alpha^2) & \dots & u(\alpha^t) \\ u(\alpha^2) & u(\alpha^3) & \dots & u(\alpha^{t+1}) \\ \vdots & \vdots & \ddots & \vdots \\ u(\alpha^{n-k-t}) & u(\alpha^{n-k-t+1}) & \dots & u(\alpha^{n-k-1}) \end{pmatrix} \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_{t-1} \end{pmatrix} = \begin{pmatrix} -u(\alpha^{t+1}) \\ -u(\alpha^{t+2}) \\ \vdots \\ -u(\alpha^{n-k}) \end{pmatrix}.$$

Lemma 5.3. *Let t be the weight of a minimum weight vector $e \in \mathbb{F}_q^n$ such that $u - e \in RS_{q,\alpha}(k)$ and consider the linear system:*

$$\begin{pmatrix} u(\alpha) & u(\alpha^2) & \dots & u(\alpha^{t'}) \\ u(\alpha^2) & u(\alpha^3) & \dots & u(\alpha^{t'+1}) \\ \vdots & \vdots & \ddots & \vdots \\ u(\alpha^{n-k-t'}) & u(\alpha^{n-k-t'+1}) & \dots & u(\alpha^{n-k-1}) \end{pmatrix} \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_{t'-1} \end{pmatrix} = \begin{pmatrix} -u(\alpha^{t'+1}) \\ -u(\alpha^{t'+2}) \\ \vdots \\ -u(\alpha^{n-k}) \end{pmatrix}.$$

1. *If $t \leq \frac{n-k}{2}$ and $t' = t$, then the linear system has a unique solution. This solution can be found as a solution to the following square system:*
$$\begin{pmatrix} u(\alpha) & u(\alpha^2) & \dots & u(\alpha^t) \\ u(\alpha^2) & u(\alpha^3) & \dots & u(\alpha^{t+1}) \\ \vdots & \vdots & \ddots & \vdots \\ u(\alpha^t) & u(\alpha^{t+1}) & \dots & u(\alpha^{2t-1}) \end{pmatrix} \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_{t-1} \end{pmatrix} = \begin{pmatrix} -u(\alpha^{t+1}) \\ -u(\alpha^{t+2}) \\ \vdots \\ -u(\alpha^{2t}) \end{pmatrix}.$$
2. *If $t \leq \frac{n-k}{2}$ and $t' = t$, then the unique solution to the system satisfies $l_0 \neq 0$.*
3. *If $t \leq \frac{n-k}{2}$ and $t' < t$, then the system has no solution.*

Now, we are ready to establish a decoding algorithm for an $RS_{q,\alpha}(k)$ code, where $n = q - 1$. We assume an input of $u \in \mathbb{F}_q^n$.

1. Let t be the minimum integer such that
$$rank \begin{pmatrix} u(\alpha) & \dots & u(\alpha^t) \\ u(\alpha^2) & \dots & u(\alpha^{t+1}) \\ \vdots & \ddots & \vdots \\ u(\alpha^{n-k-t}) & \dots & u(\alpha^{n-k-t}) \end{pmatrix} = rank \begin{pmatrix} u(\alpha) & \dots & u(\alpha^{t+1}) \\ u(\alpha^2) & \dots & u(\alpha^{t+2}) \\ \vdots & \ddots & \vdots \\ u(\alpha^{n-k-t} & \dots & u(\alpha^{n-k}) \end{pmatrix}.$$
When $t = 0$, the first matrix is the null matrix, so we consider its rank to be 0.
2. Solve the linear system
$$\begin{pmatrix} u(\alpha) & u(\alpha^2) & \dots & u(\alpha^t) \\ u(\alpha^2) & u(\alpha^3) & \dots & u(\alpha^{t+1}) \\ \vdots & \vdots & \ddots & \vdots \\ u(\alpha^t) & u(\alpha^{t+1}) & \dots & u(\alpha^{2t-1}) \end{pmatrix} \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_{t-1} \end{pmatrix} = \begin{pmatrix} -u(\alpha^{t+1}) \\ -u(\alpha^{t+2}) \\ \vdots \\ -u(\alpha^{2t}) \end{pmatrix}$$
for $l_0, \dots, l_{t-1}$. Let $\lambda_u = x^t + l_{t-1}x^{t-1} + \dots + l_1 x + l_0$.
3. Obtain as in Lemma 4.7 the interpolation polynomial f_u of u. Let d_u be the degree of u.
4. Let $\xi_0, \dots, \xi_{d_u+t}$ be the coefficients of $\lambda_u f_u$. So $\lambda_u f_u = \xi_0 + \xi_1 x + \dots + \xi_{d_u+t}x^{d_u+t}$. Let $g_c = f_u - \frac{(x^n-1)(\xi_n + \xi_{n+1}x + \dots + \xi_{d_u+t}x^{d_u+t-n})}{\lambda_u}$.
5. The output is $(g_c(1), g_c(\alpha), g_c(\alpha^2), \dots, g_c(\alpha^{n-1}))$.

Theorem 5.4. *Suppose we received $u \in \mathbb{F}_q^n$. Let t be the weight of a minimum weight vector $e \in \mathbb{F}_q^n$ such that $u - e \in RS_{q,\alpha}(k)$. If $t \leq \frac{n-k}{2}$, then the previous algorithm outputs $u - e$.*

Let's look at some examples of how we can use the algorithm to decode transmissions. We'll use the same code $C = RS_{7,5}(2)$ that we have used previously. Suppose we receive the three code words $u = 421632$, $v = 342650$, and $w = 025606$.
u has associated polynomial $4 + 2x + x^2 + 6x^3 + 3x^4 + 2x^5$. First, we multiply our parity-check matrix H by the vector u.

$$\begin{pmatrix} u(\alpha) \\ u(\alpha^2) \\ u(\alpha^3) \\ u(\alpha^4) \end{pmatrix} = \begin{pmatrix} 1 & 5 & 4 & 6 & 2 & 3 \\ 1 & 4 & 2 & 1 & 4 & 2 \\ 1 & 6 & 1 & 6 & 1 & 6 \\ 1 & 2 & 4 & 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 1 \\ 6 \\ 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ 5 \\ 4 \end{pmatrix}.$$

We have $rank \begin{pmatrix} 3 \\ 1 \\ 5 \end{pmatrix} = rank \begin{pmatrix} 3 & 1 \\ 1 & 5 \\ 5 & 4 \end{pmatrix}$. This indicates that $t = 1$. In context, $t = 1$ means that there is exactly one error in the code.
The system $u(\alpha)l_0 = -u(\alpha^2)$ gives $3l_0 = -1 = 6$, indicating that $l_0 = 2$. This means that $\lambda_u = x + 2$.
Next, we compute f_u :

$$\begin{pmatrix} 4 & 2 & 1 & 6 & 3 & 2 \end{pmatrix} \begin{pmatrix} 6 & 6 & 6 & 6 & 6 & 6 \\ 6 & 4 & 5 & 1 & 3 & 2 \\ 6 & 5 & 3 & 6 & 5 & 3 \\ 6 & 1 & 6 & 1 & 6 & 1 \\ 6 & 3 & 5 & 6 & 3 & 5 \\ 6 & 2 & 3 & 1 & 5 & 4 \end{pmatrix} = \begin{pmatrix} 3 & 0 & 3 & 2 & 6 & 4 \end{pmatrix}.$$

Therefore, $f_u = 4x^5 + 6x^4 + 2x^3 + 3x^2 + 3$.
$f_u \cdot \lambda_u = (4x^5 + 6x^4 + 2x^3 + 3x^2 + 3)(x + 2) = 4x^6 + 6x^2 + 3x + 6$.
Then $\xi_6 = 4$, so $g_c = (4x^5 + 6x^4 + 2x^3 + 3x^2 + 3) - \frac{4(x^6-1)}{x+2} = (4x^5 + 6x^4 + 2x^3 + 3x^2 + 3) - (4x^5 + 6x^4 + 2x^3 + 3x^2 + x + 5) = 6x + 5$.
So our output is $(g_c(1), g_c(5), g_c(4), g_c(6), g_c(2), g_c(3)) = (6(1)+5, 6(5)+5, 6(4)+5, 6(6)+5, 6(2)+5, 6(3)+5) = (6 + 5, 2 + 5, 3 + 5, 1 + 5, 5 + 5, 4 + 5) = (4, 0, 1, 6, 3, 2)$.

Next, let's decode $v = 342650$ with polynomial $3+4x+2x^2+6x^3+5x^4$. First, we multilpy our parity-check matrix H by the vector v.

$$\begin{pmatrix} v(\alpha) \\ v(\alpha^2) \\ v(\alpha^3) \\ v(\alpha^4) \end{pmatrix} = \begin{pmatrix} 1 & 5 & 4 & 6 & 2 & 3 \\ 1 & 4 & 2 & 1 & 4 & 2 \\ 1 & 6 & 1 & 6 & 1 & 6 \\ 1 & 2 & 4 & 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} 3 \\ 4 \\ 2 \\ 6 \\ 5 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Since this results in the null matrix, there is no error and our output is $(3, 4, 2, 6, 5, 0)$.

Next, let's decode $w = 025606$ with polynomial $2x + 5x^2 + 6x^3 + 6x^5$. First, we multiply our parity-check matrix H by the vector w.

$$\begin{pmatrix} w(\alpha) \\ w(\alpha^2) \\ w(\alpha^3) \\ w(\alpha^4) \end{pmatrix} = \begin{pmatrix} 1 & 5 & 4 & 6 & 2 & 3 \\ 1 & 4 & 2 & 1 & 4 & 2 \\ 1 & 6 & 1 & 6 & 1 & 6 \\ 1 & 2 & 4 & 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} 0 \\ 2 \\ 5 \\ 6 \\ 0 \\ 6 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 5 \\ 5 \end{pmatrix}.$$

We have $rank \begin{pmatrix} 0 & 1 \\ 1 & 5 \end{pmatrix} = rank \begin{pmatrix} 0 & 1 & 5 \\ 1 & 5 & 5 \end{pmatrix} = 2$. This indicates that $t = 2$, meaning that there are exactly two errors in the code.

The system $\begin{pmatrix}0 & 1\\ 1 & 5\end{pmatrix}\begin{pmatrix}l_0\\ l_1\end{pmatrix} = \begin{pmatrix}-5\\ -5\end{pmatrix} = \begin{pmatrix}2\\ 2\end{pmatrix}$ has solution $l_0 = 6, l_1 = 2$. This means that $\lambda_w = x^2 + 2x + 6$.
Next, we compute f_w :

$$\begin{pmatrix}0 & 2 & 5 & 6 & 0 & 6\end{pmatrix}\begin{pmatrix}6 & 6 & 6 & 6 & 6 & 6\\ 6 & 4 & 5 & 1 & 3 & 2\\ 6 & 5 & 3 & 6 & 5 & 3\\ 6 & 1 & 6 & 1 & 6 & 1\\ 6 & 3 & 5 & 6 & 3 & 5\\ 6 & 2 & 3 & 1 & 5 & 4\end{pmatrix} = \begin{pmatrix}2 & 2 & 2 & 2 & 6 & 0\end{pmatrix}.$$

Therefore, $f_w = 6x^4 + 2x^3 + 2x^2 + 2x + 2$.
$f_w \cdot \lambda_w = (6x^4 + 2x^3 + 2x^2 + 2x + 2)(x^2 + 2x + 6) = 6x^6 + 4x^3 + 4x^2 + 2x + 5$.
Then $\xi_6 = 6$, so $g_c = (6x^4+2x^3+2x^2+2x+2) - \frac{6(x^6-1)}{x^2+2x+6} = (6x^4+2x^3+2x^2+2x+2) - (6x^4-5x^3-5x^2+5x-1) = 4x + 3$.
So our output is $(g_c(1), g_c(5), g_c(4), g_c(6), g_c(2), g_c(3)) = (4(1)+3, 4(5)+3, 4(4)+3, 4(6)+3, 4(2)+3, 4(3)+3) = (4 + 3, 6 + 3, 2 + 3, 3 + 3, 1 + 3, 5 + 3) = (0, 2, 5, 6, 4, 1)$.

6 Conclusion

Maria Bras-Amorós concludes her paper with a comparison of the algorithm that she proposed to the Peterson-Gorenstein-Zierler Algorithm. This paper will not provide such an in-depth comparison. The key takeaway, however, is that Bras-Amorós's algorithm is much simpler, and is still fast when the number of errors is small. As the number of errors grows, Bras-Amorós's algorithm is not much slower than the more complex Peterson-Gorenstein-Zierler Algorithm.

Bras-Amorós's algorithm is approachable to non-experts with a relatively small amount of background information on coding theory. The definitions and theorems provided in this paper are enough to understand how the algorithm works. Teaching coding theory to university students is a daunting task, but Bras-Amorós's algorithm takes an important step towards making code more accessible.

7 References

[1] Bras-Amorós, M. (2018). A Decoding Approach to Reed-Solomon Codes from Their Definition. The American Mathematical Monthly, 125.

[2] Edwards, C. H., Penney, D. E., & Calvis, D. (2020). Differential equations & Linear Algebra. Pearson Education Limited.

[3] Irani, S. (n.d.). Introduction to Discrete Mathematics. zyBooks.

[4] Long division polynomial - definition, method, long division with monomials, Binomials. Cuemath. (n.d.). Retrieved April 25, 2023, from https://www.cuemath.com/algebra/long-division-of-polynomials/

[5] Monic polynomial. Math is Fun. (n.d.). Retrieved April 25, 2023, from https://www.mathsisfun.com/definitions/monic-polynomial.html

[6] Polynomials - what are polynomials? definition and examples. Cuemath. (n.d.). Retrieved April 25, 2023, from https://www.cuemath.com/algebra/polynomials/

[7] Reed-solomon error correcting codes from the bottom up. Electronics etc... (2022, August 7). Retrieved April 25, 2023, from https://tomverbeure.github.io/2022/08/07/Reed-Solomon.html