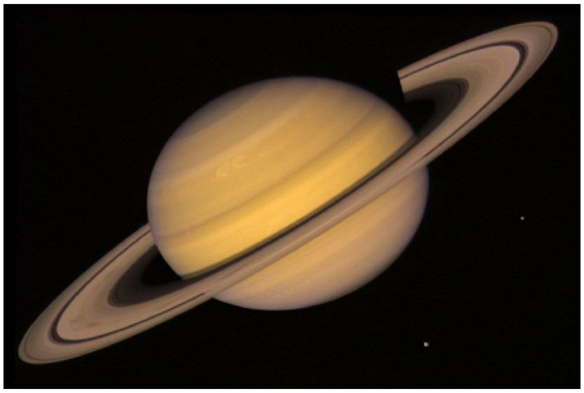
Maximal Curves and Applications to Coding Theory

Rachel Pries and Beth Malmskog

January 2011



Error-correcting codes are used to reliably transmit data in many ways, from cell phones to satellites. One way to construct good error-correcting codes is to use number theory and algebraic geometry. In this course, we will learn about curves defined over finite fields and their applications to coding theory.

1. Finite fields
2. Error-correcting codes
3. Points on curves
4. Reed-Solomon codes
5. Goppa codes and maximal curves
6. Research topics

References: Hardy and Walker, *Codes, Ciphers and Discrete Algorithms*, Discrete Mathematics and its Applications, Prentice Hall, 2009.

Walker, *Codes and Curves*, Student Mathematical Library volume 7, IAS/Park City Mathematical Subseries, American Mathematical Society, 2000.

1 Finite fields

Definition 1.1. A *field* is a commutative ring F with identity such that every non-zero element has a multiplicative inverse.

In other words, a field is a set of numbers, with two binary operations (addition and multiplication) satisfying certain associative, distributive, and commutative rules. In practice, this means we can add, subtract, multiply, and divide by anything except 0. Let's think of some examples.

The field $\mathbb{Z}/p$ In this course, the most important examples are the finite fields $\mathbb{Z}/p$, where p is a prime number. Recall that $\mathbb{Z}/p$ is the set $\{0, 1, 2, \ldots, p-1\}$ with binary operations of addition and multiplication modulo p. Another viewpoint is that $\mathbb{Z}/p$ is the set of equivalence classes of integers, under the equivalence relation of congruence modulo p.

There are several ways to show that $\mathbb{Z}/p$ is a field. In one method, given $a \not\equiv 0 \bmod p$, you show that a must have a multiplicative inverse. To do this, you solve the Diophantine equation $ax + py = 1$.

Here is an important property of the field $\mathbb{Z}/p$.

Proposition 1.2 (Fermat's Little Theorem)**.** *If $\alpha \in \mathbb{Z}/p$ then $\alpha^p - \alpha = 0$.*

Corollary 1.3. *In the polynomial ring $\mathbb{Z}/p[x]$, the polynomial $x^p - x$ factors as $\prod_{\alpha=0}^{p-1}(x - \alpha)$.*

Factorization Let k be a field (for example $\mathbb{Q}$ or $\mathbb{Z}/p$) and let $k[x]$ be the ring of polynomials with coefficients in k.

Definition 1.4. A polynomial $g(x)$ *factors in* $k[x]$ if $g(x) = h_1(x)h_2(x)$ where $h_1(x)$ and $h_2(x)$ are polynomials in $k[x]$ whose degree is less than $\deg(g(x))$. If $g(x)$ does not factor over k, then it is *irreducible in* $k[x]$ A polynomial is *monic* if its leading coefficient equals 1.

Theorem 1.5. *The polynomial ring $k[x]$ has unique factorization. If $f(x) \in k[x]$ is non-constant of degree d, then $f(x)$ has at most d roots in k.*

Proof. The main idea behind the proof is that $k[x]$ has a Euclidean algorithm and so is a principal ideal domain and a unique factorization domain. □

Constructing fields It is possible to construct fields larger than k by taking quotients of $k[x]$ or, equivalently, taking algebraic extensions of k.

Theorem 1.6. *Suppose k is a field and $g(x)$ is a polynomial which is irreducible in $k[x]$. Then the ideal $I = \langle g(x) \rangle$ of $k[x]$ is maximal and the quotient ring $k[x]/I$ is a field.*

Proof. Since $k[x]$ is a principal ideal domain, every ideal I of $k[x]$ is just the set of multiples of some polynomial $h(x)$. If $g(x)$ is irreducible, this means that the only ideals containing I are I itself and the whole ring $k[x]$. So I is maximal. With some work, this shows that every non-zero coset of the quotient ring has a multiplicative inverse. □

Corollary 1.7. *If $g(x)$ is an irreducible polynomial of degree d in $\mathbb{Z}/p[x]$, then the quotient ring $\mathbb{Z}/p[x]/\langle g(x)\rangle$ is a field of size p^d.*

Example 1.8. The polynomial $g(x) = x^2 + x + 1$ is irreducible in $R = \mathbb{Z}/2[x]$. The quotient ring is a field of size 4. The cosets are represented by $\{0, 1, \alpha, \alpha + 1\}$ where $\alpha^2 + \alpha + 1 = 0$. Every non-zero coset has an inverse since $1 \cdot 1 = 1$ and $\alpha \cdot (\alpha + 1) = \alpha^2 + \alpha \equiv -1 \equiv 1$.

Example 1.9. The polynomial $g(x) = x^3 + x + 1$ is irreducible in $R = \mathbb{Z}/2[x]$. The quotient ring is a field of size 8. The cosets are represented by

$$\{0, 1, \beta, \beta + 1, \beta^2, \beta^2 + 1, \beta^2 + \beta, \beta^2 + \beta + 1\}$$

where $\beta^3+\beta+1 \equiv 0$. The non-zero cosets form a group of size 7 under multiplication; this group is cyclic (any non-zero coset is a generator).

The theory of finite fields There is a beautiful theory about finite fields, which we could spend the whole course talking about. Here are some highlights.

Theorem 1.10.

1. *If $\mathbb{F}$ is a finite field, then the size q of $\mathbb{F}$ equals p^a for some prime p and some positive integer a.*
2. *Given $q = p^a$, the set $\mathbb{F}_q$ of roots of $x^q - x$ is a field of size q.*
3. *Given $q = p^a$, the field $\mathbb{F}_q$ is the unique field of size q.*
4. *The multiplicative group $\mathbb{F}_q^*$ is cyclic.*

Proof.

1. Main reason: it's a vector space over $\mathbb{Z}/p$.
2. Main reason: the number of roots equals p^a since the polynomial is separable. The set of roots is closed under addition/subtraction, multiplication/division.
3. Given a field $\mathbb{F}$ of size q, then $\mathbb{F}^*$ is a multiplicative group of size $q - 1$. If $\alpha \in \mathbb{F}^*$, then $\alpha^{q-1} = 1$ by Lagrange's Theorem. Then $\mathbb{F} \simeq \mathbb{F}_q$ since every element of $\mathbb{F}$ is a root of $x^q - x$.
4. By the theorem of finite abelian groups, $q - 1$ is a multiple of the exponent d of $\mathbb{F}_q^*$. Also every element of $\mathbb{F}_q^*$ is a root of $x^d - 1$; then $q - 1 \leq d$ since this polynomial has at most d roots.

□

Definition 1.11. The *Frobenius morphism* $\pi : \mathbb{F}_q \to \mathbb{F}_q$ is the function given by $\pi(\alpha) = \alpha^p$.

Theorem 1.12. *1. The Frobenius morphism is an automorphism of* $\mathbb{F}_q$.

2. The Galois group of $\mathbb{F}_q$ *over* $\mathbb{F}_p$ *is cyclic of order* a *and generated by* π.

3. $\mathbb{F}_{p^m} \subset \mathbb{F}_{p^n}$ *if and only if* m *divides* n.

Corollary 1.13. *Let* $q = p^a$ *and let* $R = \mathbb{Z}/p[x]$. *The polynomial* $x^{p^n} - x$ *factors in* R *into the product of all the monic irreducible polynomials of* R *of degree* m, *for every* m *dividing* n, *with each factor appearing once.*

Example 1.14. In $\mathbb{Z}/2[x]$, the factorization is $x^8 - x = x(x-1)(x^3+x+1)(x^3+x^2+1)$.

Problem Session 1

1. Find:

   (a) a polynomial of degree 2 which is irreducible over $\mathbb{Q}$ but factors over $\mathbb{Z}/5$.

   (b) a polynomial of degree 4 that factors over $\mathbb{Z}/2$ but has no roots.

2. (a) Find a polynomial $g(x)$ of degree 2 in $R = \mathbb{Z}/3[x]$ which has no roots.

   (b) Explain why $I = \langle g(x) \rangle$ is a maximal ideal of R.

   (c) What are representatives of the cosets in the quotient ring $K = R/I$?

   (d) Find an inverse of each coset to demonstrate explicitly that K is a field.

   (e) Find a coset $\gamma + I$ which generates K^*. In other words, if $c + I$ is a non-zero coset in R/I, then $c + I = (\gamma + I)^e$ for some exponent e.

   (f) Optional: find a ring isomorphism $\phi : R/I \to \mathbb{Z}[i]/\langle 3 \rangle$.

3. Find the three monic polynomials in $\mathbb{Z}/3[x]$ which have degree 2 and are irreducible over $\mathbb{Z}/3$. Call them g_1, g_2, g_3 and compute the product

$$(x^3 - x)g_1 g_2 g_3.$$

4. (a) Find the roots of $x^2 - 1$ in $\mathbb{Z}/8$. Why is the answer surprising?

   (b) Find two different factorizations of $x^2 - 1$ in $\mathbb{Z}/8[x]$.

Further investigation

If you like this topic, here are some more problems for you.

1. (a) For which primes p, does $x^2 + 1$ factor over $\mathbb{Z}/p$? Find some data and make a conjecture.

   (b) For which primes p, does $x^2 - 2$ factor over $\mathbb{Z}/p$? Find some data and make a conjecture.

2. (a) The polynomial $x^5 - x$ factors as $x(x-1)(x-2)(x-3)(x-4)$ over $\mathbb{Z}/5$. Use the degree 1 terms of these polynomials to prove that $4! \equiv -1 \bmod 5$.

   (b) Prove Wilson's Theorem: If p is prime, then $(p-1)! \equiv -1 \bmod p$.

2 Error Detecting and Correcting Codes

Errors often occur when data is transmitted. For example, the clerk transposes two numbers when entering the universal product code on your milk at the store, or your CD player gets bumped when you are listening to your favorite album, or some bits of data are flipped when a telescope beams the data of images of Saturn back to earth. Error detecting and error correcting codes are mathematical ways to encode information for transmission so that the receiver can at least detect and hopefully even correct errors in transmitted data.

ISBN codes One everyday example of an error detecting code is the International Standardized Book Number (ISBN). Each published book is assigned a 10 digit ISBN which is printed on the back cover of the book. For example, the ISBN for The Heart of Mathematics is 0-470-49951-1. The first nine digits are actually information about the book, while the last digit is a check digit, chosen based on the first nine digits. If any mistakes are made in entering the number, a computer can easily be programmed to detect that something is wrong and display an error message. So if a clerk who was logging inventory at a book store made a mistake entering the ISBN for a book, she could instantly know that she had goofed, and be prompted to reenter the number. How does this work? Let a_i be the i-th digit of the ISBN. Then we calculate $a'_{10} = a_1 + 2a_2 + 3a_3 + ... + 9a_9 (\text{mod } 11)$. If $0 \leq a'_{10} \leq 9$, let $a_{10} = a'_{10}$. If $a'_{10} = 10$, use $a_{10} = X$.

We can see that this will detect any single error in entering the ISBN as follows. Say that a_i is replaced by $b_i = a_i + k$ for some k. When the computer sees the first nine digits of this corrupted ISBN, it will calculate that a'_{10} should equal $a_1 + 2a_2 + 3a_3 + ... + ia_i + ik + ... + 9a_9$. This is the real value of a'_{10} plus ik. The only way that the two check digits could match is if $ik \equiv 0 \text{ mod } 11$. Since $\mathbb{Z}/11$ is a field, this is only possible if $i = 0$ or $k = 0$. Since $i \neq 0$, the digits will only match if $k = 0$, meaning no error has occurred.

Repetition Codes The first idea that we might have is to send our information twice. The receiver could check whether the two copies match. If not, there must have been a mistake. But which one has the mistake? Not so easy to tell, when the information is a string of 0s and 1s. Maybe we send each symbol 3 times, and look for differences in any of the copies. If there is a discrepancy, we could assume that the mistake most likely occurred in only one copy and go with the majority. We could correct many errors this way. However, this isn't very efficient. We have to send three times as much data as we actually want to transmit. Coding theory searches for the most efficient solutions to the problem of errors in data.

Definitions

Definition 2.1. A *code* C over and *alphabet* A of *length* n is a subset of A^n. Elements of a code are called *codewords*. We denote the number of codewords in C by $\#C$.

For the codes we consider, the alphabet will generally be a finite field $\mathbb{F}_q$ for q a prime power.

Example 2.2. Let $A = \mathbb{F}_2$, $n = 4$ and let C be the set $\{(1,0,0,1),(0,1,1,0),(1,1,1,1),(0,0,0,0)\}$.

This is an example of a *linear* code.

Definition 2.3. A function $f(x)$ is called *linear* if it possesses the following two properties:

- For all x and y in the domain of f, $f(x) + f(y) = f(x + y)$.
- For any constant α, $f(\alpha x) = \alpha f(x)$.

Example 2.4. The simplest examples of linear functions are given by linear polynomials: $f(x) = ax + b$ for constants a and b.

Definition 2.5. A *linear code* over an alphabet A of length n is a code C which possesses the following two properties:

- The all zeros vector is a codeword in C.
- For all $\vec{x} = (x_1, x_2, ..., x_n)$ and $\vec{y} = (y_1, y_2, ..., y_n)$ which are codewords in C, $\vec{x} + \vec{y} = (x_1 + y_1, x_2 + y_2, ..., x_n + y_n)$ is also a codeword in C.
- For any constant α in A and codeword x in C, $\alpha\vec{x} = (\alpha x_1, \alpha x_2, ..., \alpha x_n)$ is also a codeword in C.

In other words, a code is linear if the codewords make up a linear subspace of the vector space A^n.

Referring back to example 2.2, we can see that though $\#C = 4$, we could find all the codewords of C by knowing that C is a linear code that contains $(1,0,0,1)$ and $(0,1,1,0)$. We say that $(1,0,0,1)$ and $(0,1,1,0)$ are *generators* of C, i.e. they generate C as a vector space. To write down C as compactly as possible, we might use the matrix $\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$.

Definition 2.6. The *dimension* of a linear code C is the minimal number of codewords that are needed to generate all the codewords of C using definition 2.5. The dimension of a code is its dimension as a linear subspace. A minimal set of codewords that will generate C is called a *basis* for C. The length n and dimension k of a code are called its *parameters*. A $k \times n$ matrix with rows consisting of a basis of C is called a *generator matrix* for C.

Distances in Codes It will help us to think about codes geometrically. We can think of the codewords as points in the space A^n. How do we understand the distance between codewords?

Definition 2.7. For $\vec{x} = (x_1, x_2, ..., x_n)$ and $\vec{y} = (y_1, y_2, ..., y_n)$ in A^n, the *Hamming distance* between $\vec{x}$ and $\vec{y}$ is defined to be

$$d(\vec{x}, \vec{y}) = \#\{i : x_i \neq y_i\}.$$

Let $\vec{0}$ be the all zeros vector of length n. The *Hamming weight* of x is defined to be $wt(\vec{x}) = d(\vec{x}, \vec{0}) = \#\{i : x_i \neq 0\}$.

Definition 2.8. The minimum distance of C is defined as

$$d_{min}(C) = \min\{wt(\vec{x}) : \vec{x} \in C\}.$$

It's straight forward to see that the minimum distance of a code is equal to the minimum weight of any codeword of C excluding $\vec{0}$. If a code C has minimum distance d, then for any $\vec{x}$ in C, there is no codeword $\vec{y}$ in C so that $d(\vec{x}, \vec{y}) < d$. We can think of this as a ball of radius $\frac{d-1}{2}$ centered at each codeword $\vec{x}$ such that $\vec{x}$ is the closest codeword to any point of A^n in this ball.

Definition 2.9. A *ball* $B_r(\vec{x})$ of radius r about a codeword $\vec{x}$ is the set of all $\vec{y}$ in A^n so that $d(\vec{x}, \vec{y}) \leq r$.

Say that we are using C to encode our information and we receive a transmission x', which is not a codeword. If $\vec{x}'$ is in $B_{\frac{d-1}{2}}$, that is, if $d(\vec{x}, \vec{x}') \leq \frac{d-1}{2}$, we can assume that the original codeword was likely to be x. Thus the minimum distance of a code determines the number of errors that the code can correct. A code with minimum distance d can correct up to $\frac{d-1}{2}$ errors. The minimum distance also tells us about a code's ability to detect errors, even if it can't correct them properly. If the smallest distance between two codewords is d, then a codeword would need to be affected by at least d errors before the transmission could appear to be a different valid codeword. So a computer could be programmed to notify the recipient that errors occurred as long as there are $d - 1$ or fewer of them.

Bounds on codes We want error correcting codes to be efficient, meaning that we want to transmit as few extra symbols as possible. For a linear code of length n and dimension k, we can think of each codeword as having k symbols of information and n total symbols. We would like k to be large with respect to n. We also want error correcting codes to be able to catch and correct a lot of errors. The longer the codewords are, the more errors we would be likely to see. If the minimum distance of the code is d we would like d to also be large with respect to n. As you might guess, there is some trade-off between k and d.

Theorem 2.10. *(The Singleton Bound) Let C be a linear code over $\mathbb{F}_q$ of length n, dimension k, and minimum distance d.*

$$d \leq n - k + 1.$$

Proof of the theorem. Let W be the linear subspace of $\mathbb{F}_q^n$ which has zeros in the d-th through n-th components. That is, let

$$W = \{\vec{x} = (x_1, x_2, ..., x_n) \in \mathbb{F}_q^n : x_d = x_{d+1} = ... = x_n = 0\}.$$

We can see that W is a $(d-1)$-dimensional subspace of $\mathbb{F}_q^n$, and $wt(\vec{x}) \leq (d-1)$ for any $\vec{x}$ in W, meaning x is not in C. Therefore $W \cap C = \{\vec{0}\}$. Let

$$W + C = \{\vec{w} + \vec{c} : \vec{w} \in W, \vec{c} \in C\}.$$

The dimension of $W + C$ is $k + d - 1$, so $k + d - 1 \leq n$. Therefore $d \leq n - k + 1$.

Problem Session 2

1. The first nine digits of the ISBN for a book are 0-312-59034. What is the check digit?

2. You have 9 stacks of 9 coins, all of which appear to be identical. One of these stacks is made up of all fake coins, while the rest of them are real gold. The real coins weigh 10 grams each, while the counterfeits weigh 9 grams each. You have a scale that will tell you the mass of any combination of the coins, but you want to be clever and find out which stack contains the fake coins in the least possible number of weighings. How do you do it?

3. Consider the code C over $\mathbb{F}_2$ with the following generator matrix, also called C:

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

   (a) Find the parameters n and k of C.

   (b) What is the minimum distance d of C?

   (c) You receive the transmission $(1,0,0,1,1,1,0)$. Is this a codeword? If not, what was the intended codeword?

   (d) How many codewords are there in C?

   (e) A code that meets the Singleton bound, that is a code for which $d = n - k + 1$, is called a *Maximum Distance Separable* code (MDS code). Is C an MDS code?

   (f) How many errors can C correct? How many can it detect?

4. Prove that the minimum distance of a linear code C is the same as the minimum weight of any codeword in C except $\vec{o}$.

5. How many codewords are there in a linear code of dimension k over $\mathbb{F}_q$?

6. Given a linear code C, we now have experience finding the length, dimension, minimum distance, and number of codewords in C. But given arbitrary length, dimension, minimum distance, and number of codewords, does a code C exist with these parameters? Not always.

   Definition 2.11. The quantity $A_q(n,d)$ is the maximum value of M such that there is a linear code of length n with M codewords and minimum distance d.

   Use the Singleton bound and exercise 5 to give an upper bound on $A_q(n,d)$ for linear codes in terms of n, q, and d.

7. Recall that, if our alphabet is $\mathbb{F}_q$, we defined the ball about a point $\vec{x}$ of radius r to be $B_r(\vec{x}) = \{\vec{y} \in \mathbb{F}_q^n : d(\vec{x}, \vec{y}) \leq r\}$.

   Definition 2.12. The quantity $V_q(n, r)$ is the number of points of $\mathbb{F}_q^n$ in the ball of radius r centered at any point of $\mathbb{F}_q^n$.

   Prove that

$$V_q(n, r) = \sum_{i=0}^{r} \binom{n}{i} (q-1)^i$$

   where

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

   is the binomial coefficient that counts the number of (unordered) different ways to choose k things from a set of n.

The previous two exercises give us the notation to state and understand the *Gilbert-Varshamov Bound*:

$$A_q(n, d) \geq \frac{q^n}{V_q(n, d-1)}.$$

3 Points on curves

Let k be a field. In these talks, k will be a finite field $\mathbb{F}_q$ where $q = p^a$ where p is prime, for example, $k = \mathbb{Z}/p$.

Planar curves Informally, a planar curve is the set of points (x, y) satisfying a polynomial equation $f(x, y) \in k[x, y]$ in two variables.

Example 3.1. (a) Rational curve $y = h(x)$.

(b) Hyperelliptic curve $y^2 = h(x)$.

(c) Hermitian curve $y^p + y = x^{p+1}$.

Points on curves

Example 3.2. Recall that $\mathbb{F}_4 = \{0, 1, \alpha, \alpha + 1\}$ where $2 \equiv 0$ and $\alpha^2 + \alpha + 1 \equiv 0$. Let's find the $\mathbb{F}_4$-points of the curve $y^2 + y = x^3$.

y	0	1	α	$\alpha + 1$
$y^2 + y$	0	0	1	1

x	0	1	α	$\alpha + 1$
x^3	0	1	1	1

So there are 8 points: $(0, 0)$, $(0, 1)$, $(1, \alpha)$, $(1, \alpha+1)$, (α, α), $(\alpha, \alpha+1)$, $(\alpha+1, \alpha)$, $(\alpha + 1, \alpha + 1)$.

Points at infinity One drawback to working with planar curves is that they are not projective (complete, compact). To fix this, we need to study points at infinity. The points at infinity are in bijection with slopes of asymptotes to the curve. Here is the method to find them.

Step 1: *Homogenize* the equation by capitalizing x and y and adding powers of a new variable Z.

Step 2: Set $Z = 0$ and find the conditions this places on X and Y.

Step 3: The points at infinity are solutions $[X, Y, 0]$ to the homogenized equation, with X, Y not both zero, up to scalar equivalence.

Example 3.3. There is one point at ∞ on the curve $y^2 + y = x^3$. The homogenized equation is $Y^2Z + YZ^2 = X^3$. If $Z = 0$, then $X = 0$. So the solutions are $[0, Y, 0]$. Up to scalar equivalence, there is one point at infinity, namely $[0, 1, 0]$.

Why does this work? To describe it precisely, we would define the projective plane $\mathbb{P}^2$. Its points are of the form $[X, Y, Z]$ with $X, Y, Z \in k$ not all 0, where two points are equivalent if one is a scalar multiple of the other. A projective curve in $\mathbb{P}^2$ is the set of equivalence classes of points $[X, Y, Z]$ satisfying a homogenous equation $F(X, Y, Z)$. The affine plane is contained in the projective plane $\mathbb{A}^2 \subset \mathbb{P}^2$. Given a planar curve $f(x, y)$, there is a unique projective curve $F(X, Y, Z)$ whose restriction to $\mathbb{A}^2$ is f.

Singular points

Definition 3.4. A planar curve $f(x, y)$ is *singular* at a point on the curve if both partial derivatives of f vanish at that point. A projective curve $F(X, Y, Z)$ is *singular* at a point on the curve if all three partial derivatives vanish at that point.

Example 3.5. There are no singular points on the curve $y^2+y = x^3$ because $f_y = 1$ everywhere. For the projective curve $Y^2Z + YZ^2 = X^3$, the partial derivatives are $F_X = -3X^2 \equiv X^2$, $F_Y = Z^2$ and $F_Z = Y^2$. These partial derivatives are simultaneously zero only at $[0, 0, 0]$ which is not a valid point of projective space.

The genus The genus of a complex Riemann surface equals the number of its 'handles' and is a topological invariant. It is also possible to define the genus of a smooth projective curve defined over $\overline{\mathbb{F}}_p$; it is the dimension of the vector space of holomorphic 1-forms. Naively speaking, it measures how complicated the curve is.

Lemma 3.6. *(a) The genus of a rational curve $y = h(x)$ is 0.*

(b) The genus of a hyperelliptic curve $y^2 = h(x)$ where $\deg(h(x)) = d$ is $\lfloor (d-1)/2 \rfloor$.

(c) Plücker formula: If C is a smooth projective curve of degree d in $\mathbb{P}^2$ then $g = (d-1)(d-2)/2$.

The Riemann-Hurwitz formula is a good method to find the genus of a curve.

Problem Session 3:

1. Let's count the $\mathbb{F}_9$ points on the curve $y^3 + y = x^4$. Write $\mathbb{Z}/3 = \{0, \pm 1\}$ and $\mathbb{F}_9 = \{a + b\gamma \mid a, b \in \mathbb{Z}/3\}$ where $\gamma^2 = -1$.

   (a) Fill in these tables:

x	0	± 1	$\pm\gamma$	$\pm(1+\gamma)$	$\pm(1-\gamma)$
x^4					

y	0	1	-1	γ	$\gamma+1$	$\gamma-1$	$-\gamma$	$-\gamma+1$	$-\gamma-1$
y^3+y									

   (b) Use the tables to find all the solutions (x, y) to $y^3+y = x^4$ when $x, y \in \mathbb{F}_9$.

   (c) Find the points at infinity for the curve $y^3 + y = x^4$.

   (d) How many $\mathbb{F}_9$-points does the curve $y^3 + y = x^4$ have?

   (e) Does the curve $y^3 + y = x^4$ have any singular points?

2. Consider the curve defined by the equation $y^2 = x^3 + x$.

   (a) Give the homogenous form of the above equation and find the points at infinity.

   (b) Show that the curve is smooth as long as the characteristic is not 2.

   (c) Determine the genus of the curve.

3. Consider the curves defined by $y^2 = x^3 + x^2$ and $y^2 = x^3$.

   (a) Draw the graphs of the curves in $\mathbb{R}^2$.

   (b) Determine the singular points of the curves.

4 Reed-Solomon codes

Reed-Solomon codes were invented in 1960 at MIT Lincoln Lab. At that time, technology was too weak to implement them. Their first uses (in the early 1980s) were for digital photos for Voyager space probe and compact disks.

Notation 4.1. Let $q = p^n$ be a prime power and label the non-zero elements of $\mathbb{F}_q$ as $\alpha_1, \ldots, \alpha_{q-1}$. Let k be such that $1 \leq k \leq q-1$. Let L_{k-1} be the set of polynomials $g(x)$ with $\deg(g) \leq k-1$. Given $f \in L_{k-1}$, let $f(\vec{\alpha}) = (f(\alpha_1), \ldots, f(\alpha_{q-1}))$.

Definition 4.2. The Reed-Solomon code $\mathrm{RS}(k,q)$ is the subset of $\mathbb{F}_q^{q-1}$ consisting of all vectors $f(\vec{\alpha})$ for $f \in L_{k-1}$.

Lemma 4.3. *(i) L_{k-1} is a vector space over $\mathbb{F}_q$ of dimension k.*

(ii) $\mathrm{RS}(k,q)$ is a linear code over $\mathbb{F}_q$ of length $q-1$ and dimension k.

(iii) The number of codewords is q^k.

Proof. (i) A basis for L_{k-1} over $\mathbb{F}_q$ is $\{1, x, x^2, \ldots, x^{k-1}\}$.

(ii) Suppose $f_1(\vec{\alpha})$ and $f_2(\vec{\alpha})$ are in $\mathrm{RS}(k,q)$. Then $f_1(x)$ and $f_2(x)$ are in L_{k-1}. By part (i), L_{k-1} is a vector space over $\mathbb{F}_q$. So if $c \in \mathbb{F}_q$, then $cf_1 + f_2 \in L_{k-1}$. Then $cf_1(\vec{\alpha}) + cf_2(\vec{\alpha}) = (cf_1 + f_2)(\vec{\alpha}) \in \mathrm{RS}(k,q)$. Thus $\mathrm{RS}(k,q)$ is a linear code over $\mathbb{F}_q$. The dimension of $\mathrm{RS}(k,q)$ is the same as the dimension of L_{k-1}. A basis for $\mathrm{RS}(k,q)$ is given by the vectors $(\alpha_1^i, \ldots, \alpha_{q-1}^i)$ for $1 \leq i \leq k-1$.

□

Example 4.4. Let $q = 5$ and $k = 3$. Let $\alpha_1 = 1, \ldots, \alpha_4 = 4$. A basis for L_2 is $\{1, x, x^2\}$. Let $f_1(x) = 1$, let $f_2(x) = x$, and $f_3(x) = x^2$. Then $f_1(\vec{\alpha}) = (1,1,1,1)$, and $f_2(\vec{\alpha}) = (1,2,3,4)$, and $f_3(\vec{\alpha}) = (1,4,4,1)$. These three vectors are a basis for $\mathrm{RS}(2,5)$. This is a code over $\mathbb{Z}/5$ with length 4 and dimension 3.

Remark 4.5. Notice that L_{k-1} is the vector space of functions on $\mathbb{P}^1$ whose only poles are at ∞ and such that the order of the pole at ∞ is at most $k-1$. The entries of a codeword are the values of the function at the (non-zero) points of $\mathbb{P}^1$.

Transmitting data One interesting thing about the Reed-Solomon code is that all the entries are treated equally. Unlike the ISBN code or the $(7,4)$ code, there are no 'check digits'. The data does not appear as part of the transmitted message. So how does it work?

Transmitting data:

Definition 4.6. If $\vec{c} = (c_0, \ldots, c_{k-1})$ is the data, then let $f_{\vec{c}}(x) = \sum_{i=0}^{k-1} c_i x^i$ and the code word is $f_{\vec{c}}(\vec{\alpha})$.

Example 4.7. Suppose $q = 5$ and $k = 3$. Then we can transmit three pieces of data, let's say (c_0, c_1, c_2). First we create a function $f(x) = c_0 + c_1 x + c_2 x^2$. Then the code word is $(f(1), f(2), f(3), f(4))$. Specifically, if the data is $(4, 0, 1)$ then $f(x) = x^2 + 4$ and the code word is $(0, 3, 3, 0)$.

This can be implemented easily using linear algebra.

Definition 4.8. The generator matrix $M_{k,q}$ for $\mathrm{RS}(k, q)$ is the matrix with $q - 1$ columns and k rows constructed as follows: let $f_1(x) = 1, f_2(x) = x, \ldots, f_k(x) = x^{k-1}$. The jth row of $M_{k,q}$ is the codeword $f_k(\vec{\alpha})$. Given a data vector $\vec{c} = (c_0, \ldots, c_{k-1})$, the code word is $\vec{c} M_{k,q}$.

Example 4.9.

$$M_{3,5} = \begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 4 & 4 & 1 \end{matrix}$$

Interpreting data: Suppose the codeword is $\vec{b} = (b_1, \ldots, b_{q-1})$. We need to find a function $f(x)$ with degree at most $k-1$ such that $f(\vec{\alpha}) = \vec{b}$, i.e., such that $f(\alpha_i) = b_i$. Then the data is given by the coefficients of $f(x)$.

Lemma 4.10. *The data vector $\vec{c}$ is the solution to the linear system $\vec{c} M_{k,q} = \vec{b}$.*

Specifically, we can find the data vector $\vec{c}$ using row reduction.

Lemma 4.11. *If the linear system is inconsistent, then there has been an error in transmission.*

Example 4.12. There is no vector in $\mathrm{RS}(5, 3)$ with weight 1. To see this, suppose $f(\vec{\alpha})$ is a vector in $\mathrm{RS}(5, 3)$ with length 4 and weight 1; this means exactly three entries are zero, e.g., $(0, 0, 0, 2)$. Then $f(x)$ has three roots in $\mathbb{Z}/5$. This is impossible since $\deg(f) \leq 2$.

The minimal distance of a Reed-Solomon code Recall that the minimal distance of a code is the smallest non-zero Hamming distance between two codewords. This measures how many entries are different between the codewords. It determines the number of errors in transmission that can be detected and corrected. The distance of a Reed-Solomon code is optimal given the fixed length $q - 1$ and dimension k.

Theorem 4.13. *The Reed-Solomon code $\mathrm{RS}(k, q)$ has distance $d = q - k$.*

Proof. By the Singleton bound, $d \leq n-k+1$ where n is the length of the codewords. Since $n = q-1$, this implies $d \leq q-k$.

To prove that $d \geq q-k$, recall that the minimal distance of a linear code is the same as the minimal weight. The weight of a codeword is the number of its entries which are non-zero. So we need to show that if $f(\vec{\alpha})$ is a non-zero codeword, then the number of its entries which are non-zero is at least $q-k$. In other words, we need to show that the number of roots of $f(x)$ is at most $(q-1)-(q-k) = k-1$. This is true since $f(x)$ has degree at most $k-1$. □

The Reed-Solomon codes are very good codes. For fixed q and k, the distance of the code $\mathrm{RS}(k,q)$ is optimal. One drawback of the Reed-Solomon codes is that, once q is fixed, the length of the codewords is fixed at $q-1$ and the dimension is bounded by $q-1$. Next time we will look for codes where the distance and dimension can be large relative to q.

Problem Session 4:

1. Let's investigate the Reed-Solomon code RS$(2, 5)$.

   (a) What is a basis for the codewords? How many codewords are there? What is the matrix $M_{2,5}$?

   (b) If the data is $\vec{c} = (1, 3)$, what is the codeword?

   (c) If the codeword is $\vec{b} = (1, 0, 4, 3)$, what is the data?

   (d) If the codeword is $\vec{b} = (1, 0, 2, 4)$, show that an error occurred in transmission. What is the best guess for the data vector $\vec{c}$?

   (e) What is the distance invariant for RS$(2, 5)$? How many errors can this code detect? How many errors can this code correct?

2. What happens to RS(k, q) if we let $k \geq q$? Find a basis for RS$(5, 3)$.

3. If I want a Reed-Solomon code with at least 200 codewords that will correct 2 errors, what could the parameters be? What is the smallest choice of q?

4. This exercise helps to prove the assertion in the proof of Theorem 4.3 that the dimension of RS(k, q) is k.

   (a) Given $\alpha_1 \neq \alpha_2 \in \mathbb{F}_q$, find f_1, f_2 polynomials of degree 1 such that

   $$f_1(\alpha_1) = 1, \qquad f_2(\alpha_1) = 0,$$

   and

   $$f_1(\alpha_2) = 0, \qquad f_2(\alpha_2) = 1.$$

   (b) Given $\alpha_1 \neq \alpha_2 \neq \alpha_3$, find f_1 a polynomial of degree 2 such that

   $$f_1(\alpha_1) = 1, \qquad f_1(\alpha_2) = 0, \qquad f_1(\alpha_3) = 0.$$

   (c) Explain why RS(k, q) contains codewords beginning with $(1, 0, 0, 0, ...)$, $(0, 1, 0, 0, ...)$, etc.

   (d) Explain why RS(k, q) has dimension $\geq k$.

5 Goppa codes and maximal curves

In 1977, Goppa invented new error-correcting codes using algebraic geometry. The strategy is to use a curve C and a set S of points on C defined over a finite field $\mathbb{F}$. The codewords are constructed by evaluating functions on C at the points of S.

Definition 5.1. Let C be a smooth projective curve over $\mathbb{F}_q$. Let $D = rP_\infty$ where P_∞ is the point at infinity of C and r is a natural number. Let $S = \{\alpha_1, \ldots, \alpha_n\}$ be a set of distinct points of C defined over $\mathbb{F}_q$, not including P_∞. (More generally, D is a divisor of C, and S is a set of points disjoint from the support of D.) Let $L(D)$ be the set of functions on C having poles only at P_∞ such that the order of the pole at ∞ is at most r.

For $f \in L(D)$, the codeword is $f(\vec{S}) = (f(\alpha_1), \ldots, f(\alpha_n))$. The Goppa code is

$$G(C, S, D) = \{f(\vec{S}) \mid f \in L(D)\}.$$

The Reed-Solomon code evaluates functions on points on a line $y = 0$.

Theorem 5.2. *Let g be the genus of C. If $2g - 2 < r < n$, then the Goppa code $G(C, S, D)$ is a linear code of length n, dimension $k = r + 1 - g$ and minimal distance d for some $d \geq n - r$.*

Proof. The code is linear since $L(D)$ is linear and has length n by definition. The Riemann-Roch formula is needed to show that $\dim(L(D)) = k$. The lower bound on d comes from an upper bound on the number of zeros of a function whose poles have order at most r. □

Example 5.3. Recall that $\mathbb{F}_4 = \{0, 1, \alpha, \alpha + 1\}$ where $2 \equiv 0$ and $\alpha^2 + \alpha + 1 \equiv 0$. Let C be the curve $Y^2Z + YZ^2 = X^3$ and let P_∞ be the point at ∞ of C. Other than P_∞, there are eight $\mathbb{F}_4$-points on C, namely $\alpha_0 = (0, 0)$, $\alpha_1 = (0, 1)$, $\alpha_2 = (1, \alpha)$, $\alpha_3 = (1, \alpha + 1)$, $\alpha_4 = (\alpha, \alpha)$, $\alpha_5 = (\alpha, \alpha + 1)$, $\alpha_6 = (\alpha + 1, \alpha)$, $\alpha_7 = (\alpha + 1, \alpha + 1)$. Let $S = \{\alpha_1, \ldots, \alpha_7\}$.

In $\mathbb{P}^2$, the line $Z = 0$ intersects C three times at P_∞; the line $Y = 0$ intersects C three times at α_0; The line $X = 0$ intersects C at α_0, α_1 and P_∞.

A basis for $L(3P_\infty)$ is $\{1, X/Z, Y/Z\}$. The function $f_1 = 1$ gives the codeword

$$(1, 1, 1, 1, 1, 1, 1).$$

The function $f_2 = X/Z$ gives the codeword

$$(0, 1, 1, \alpha, \alpha, \alpha + 1, \alpha + 1).$$

The function $f_2 = Y/Z$ gives the codeword

$$(1, \alpha, \alpha + 1, \alpha, \alpha + 1, \alpha, \alpha + 1).$$

This is a Goppa code of length 7, dimension 3, and $d \geq 4$. In fact, $d = 5$.

For fixed q, to optimize a Goppa code, we would like to have n large in comparison with g. In other words, we need C to have a lot of points defined over $\mathbb{F}_q$ in comparison with g. But there are limits.

Theorem 5.4 (Hasse-Weil bound)**.** *If C is a smooth projective curve of genus g, then the number of points of C defined over $\mathbb{F}_q$ is at most $q+1+2g\sqrt{q}$.*

Definition 5.5. A curve C is *maximal over* $\mathbb{F}_q$ if it realizes the Hasse-Weil bound over $\mathbb{F}_q$.

Here is an example of a maximal curve.

Definition 5.6. Given a prime power q, the Hermitian curve H_q is the smooth projective curve with affine equation $y^q + y = x^{q+1}$. Its homogenous equation is $Y^qZ + YZ^q = X^{q+1}$.

Theorem 5.7. *(i) The genus of H_q is $q(q-1)/2$.*

(ii) The number of $\mathbb{F}_{q^2}$ points of H_q is q^3+1.

(iii) The curve H_q is maximal over $\mathbb{F}_{q^2}$.

Proof. (i) The partial derivatives of $Y^qZ + YZ^q = X^{q+1}$ are $F_X = -X^q$, $F_Y = Z^q$ and $F_Z = Y^q$. These are simultaneously zero only at $[0,0,0]$ which is not a valid point of projective space. So H_q is a smooth projective curve of degree $d = q+1$ in $\mathbb{P}^2$. By the Plücker formula, the genus of H_q is $g = q(q-1)/2$.

(Another way to prove this is to consider the map $\phi : H_q \to \mathbb{P}^1$ given by $\phi([X : Y : Z]) = [Y : Z]$. This map has degree $q+1$ unless $X = 0$ or $Z = 0$; this means there are $q+1$ points of $\mathbb{P}^1$ above which the map is 1-to-1. Then the genus can be computed using the Riemann-Hurwitz formula.)

(ii) If $Z = 0$ then $X = 0$, and so $[0 : 1 : 0]$ is the only point at infinity on H_q.

Consider the trace map $\mathrm{Tr} : \mathbb{F}_{q^2} \to \mathbb{F}_q$ with formula $\mathrm{Tr}(y) = y^q + y$. It is a surjective q-to-1 additive map. Also consider the norm map $\mathrm{N} : \mathbb{F}_{q^2}^* \to \mathbb{F}_q^*$ with formula $\mathrm{N}(x) = x^q \cdot x$. It is a surjective $q+1$-to-q multiplicative map.

Suppose $x, y \in \mathbb{F}_{q^2}$ with $y^q + y = x^{q+1}$. Then $\mathrm{Tr}(y) = \mathrm{N}(x) \in \mathbb{F}_q$. If $\mathrm{N}(x) = 0$, then there is one choice for x, namely $x = 0$, and q choices for y. If $\mathrm{N}(x) \neq 0$, then there are $q+1$ choices for x and q choices for y. This gives $(q-1)(q+1)q + q + 1 = q^3 + 1$ points defined over $\mathbb{F}_{q^2}$.

(iii) The number N of points of H_q defined over $\mathbb{F}_{q^2}$ is at most $q^2 + 1 + 2gq = q^3 + 1$ by the Hasse-Weil bound. Thus the upper bound is achieved.

□

Tsfasman, Vladut, and Zink used modular curves to find a sequence of Goppa codes with asymptotically better parameters than any earlier known codes.

Problem session 5:

1. (From Walker, problems 5.15 and 6.5) Let C be the projective elliptic curve with equation
$$Y^2Z + YZ^2 = X^3 + Z^3$$
defined over the field $\mathbb{F}_2$.

   (a) Show that C is smooth and has genus 1.

   (b) Find the set S of points of C defined over $\mathbb{F}_4$.

   (c) Show that the following functions have poles only at ∞ and find the order of the poles at ∞:
$$1,\ X/Z,\ Y/Z,\ X^2/Z^2,\ XY/Z^2.$$

   (d) Find a basis of the Goppa code $G(C, S, 5P_\infty)$.

   (e) What are the invariants of this code?