An introduction to Algebraic Geometry codes

Elena Berardini

Eindhoven University of Technology

ICTLab Group meeting
4^{th} April 2023

Personal webpage: http://www.elenaberardini.it/
Questions? e.berardini@tue.nl

Road map

1. Linear codes and Reed–Solomon codes
2. Algebraic geometry codes

1 Linear codes and Reed–Solomon codes

2 Algebraic geometry codes

Linear codes

Let $\mathbb{F}_q$ denote the finite field with q elements.
A linear code C on $\mathbb{F}_q$ of length n is a vector subspace of $\mathbb{F}_q^n$. Let k be its dimension.

A G matrix of C is a matrix whose rows form a basis of C. (often taken in row-reduced echelon form)

Linear codes

Let $\mathbb{F}_q$ denote the finite field with q elements.
A linear code C on $\mathbb{F}_q$ of length n is a vector subspace of $\mathbb{F}_q^n$. Let k be its dimension.

A G matrix of C is a matrix whose rows form a basis of C. (often taken in row-reduced echelon form)

Let $\boldsymbol{x} \in C$. The weight of the word x is given by $\omega(\boldsymbol{x}) = \#\{i \in \{1, \ldots, n\} \mid x_i \neq 0\}$.
Ex: the weight of $(1, 0, 2, 0, 0, 0) \in \mathbb{F}_3^6$ is 2.

Linear codes

Let $\mathbb{F}_q$ denote the finite field with q elements.
A linear code C on $\mathbb{F}_q$ of length n is a vector subspace of $\mathbb{F}_q^n$. Let k be its dimension.

A G matrix of C is a matrix whose rows form a basis of C. (often taken in row-reduced echelon form)

Let $\boldsymbol{x} \in C$. The weight of the word x is given by $\omega(\boldsymbol{x}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq 0\}$.
Ex: the weight of $(1, 0, 2, 0, 0, 0) \in \mathbb{F}_3^6$ is 2.

Let $\boldsymbol{x}$, $\boldsymbol{y} \in C$. The Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined by

$$d(\boldsymbol{x}, \boldsymbol{y}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq y_i\} = \omega(\boldsymbol{x} - \boldsymbol{y}).$$

Linear codes

Let $\mathbb{F}_q$ denote the finite field with q elements.
A linear code C on $\mathbb{F}_q$ of length n is a vector subspace of $\mathbb{F}_q^n$. Let k be its dimension.

A G matrix of C is a matrix whose rows form a basis of C. (often taken in row-reduced echelon form)

Let $\boldsymbol{x} \in C$. The weight of the word x is given by $\omega(\boldsymbol{x}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq 0\}$.
Ex: the weight of $(1, 0, 2, 0, 0, 0) \in \mathbb{F}_3^6$ is 2.

Let $\boldsymbol{x}, \boldsymbol{y} \in C$. The Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined by

$$d(\boldsymbol{x}, \boldsymbol{y}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq y_i\} = \omega(\boldsymbol{x} - \boldsymbol{y}).$$

The minimum distance of the code C is defined by $d_{min}(C) \overset{\text{def}}{=} \min_{\substack{\boldsymbol{x}, \boldsymbol{y} \in C \\ \boldsymbol{x} \neq \boldsymbol{y}}} d(\boldsymbol{x}, \boldsymbol{y}) = \min_{\boldsymbol{x} \in C \setminus \{0\}} \omega(\boldsymbol{x})$.

Linear codes

Let $\mathbb{F}_q$ denote the finite field with q elements.
A linear code C on $\mathbb{F}_q$ of length n is a vector subspace of $\mathbb{F}_q^n$. Let k be its dimension.

A G matrix of C is a matrix whose rows form a basis of C. (often taken in row-reduced echelon form)

Let $\boldsymbol{x} \in C$. The weight of the word x is given by $\omega(\boldsymbol{x}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq 0\}$.
Ex: the weight of $(1, 0, 2, 0, 0, 0) \in \mathbb{F}_3^6$ is 2.

Let $\boldsymbol{x}, \boldsymbol{y} \in C$. The Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined by

$$d(\boldsymbol{x}, \boldsymbol{y}) = \#\{i \in \{1, \dots, n\} \mid x_i \neq y_i\} = \omega(\boldsymbol{x} - \boldsymbol{y}).$$

The minimum distance of the code C is defined by $d_{min}(C) \overset{\text{def}}{=} \min_{\substack{\boldsymbol{x}, \boldsymbol{y} \in C \\ \boldsymbol{x} \neq \boldsymbol{y}}} d(\boldsymbol{x}, \boldsymbol{y}) = \min_{\boldsymbol{x} \in C \setminus \{0\}} \omega(\boldsymbol{x})$.

$[n, k, d]_q$–code: code of length **n**, dimension **k** and minimum distance **d**.

$$\left.\begin{array}{r} \text{dimension} \leftrightarrow \text{information rate} \\ \text{minimum distance} \leftrightarrow \text{correction capacity} \end{array}\right\} \quad k + d \leqslant n + 1 \text{ Singleton, 1964}$$

About parameters of linear codes

For an $[n, k, d]$–code C, we define its *(transmission) rate* $\kappa \stackrel{\text{def}}{=} \frac{k}{n}$ and its *relative distance* $\delta \stackrel{\text{def}}{=} \frac{d}{n}$.

"Good" code : κ and δ close to 1.

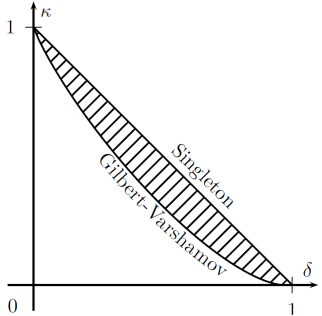
About parameters of linear codes

For an $[n,k,d]$–code C, we define its *(transmission) rate* $\kappa \stackrel{\text{def}}{=} \frac{k}{n}$ and its *relative distance* $\delta \stackrel{\text{def}}{=} \frac{d}{n}$.

"Good" code : κ and δ close to 1.

Compromises:

- **Singleton bound**: $\delta + \kappa \leq 1 + \frac{1}{n}$.
- **Gilbert-Varshamov "bound"**:
  With fixed q and $n \to +\infty$,
  $\sup_{C\ q\text{–ary}} \{\kappa(C) \mid \delta(C) = \delta\} \geq 1 - H_q(\delta)$ where H_q is the entropy function defined by
  $H_q(\delta) \stackrel{\text{def}}{=} \delta \log_q(q-1) - \delta \log_q \delta - (1-\delta)\log_q(1-\delta)$.



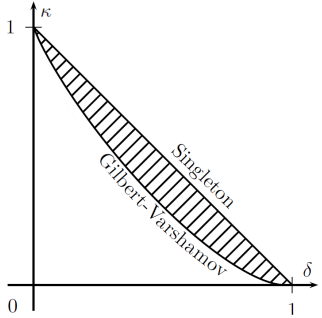
About parameters of linear codes

For an $[n,k,d]$–code C, we define its *(transmission) rate* $\kappa \stackrel{\text{def}}{=} \frac{k}{n}$ and its *relative distance* $\delta \stackrel{\text{def}}{=} \frac{d}{n}$.

"Good" code : κ and δ close to 1.

Compromises:

- **Singleton bound**: $\delta + \kappa \leq 1 + \frac{1}{n}$.
- **Gilbert-Varshamov "bound"**:
  With fixed q and $n \to +\infty$,
  $\sup_{C\ q\text{–ary}} \{\kappa(C) \mid \delta(C) = \delta\} \geq 1 - H_q(\delta)$ where H_q is the entropy function defined by
  $H_q(\delta) \stackrel{\text{def}}{=} \delta \log_q(q-1) - \delta \log_q \delta - (1-\delta)\log_q(1-\delta)$.



A random (linear) code of length n and dimension k satifies $\frac{k}{n} \simeq 1 - H_q(\frac{d}{n})$, with probability going to 1 when $n \to \infty$.

Reed–Solomon codes

Let $\mathbb{F}_q[X]_{<k}$ be the set of univariate polynomials with coefficients in $\mathbb{F}_q$ and degree $< k$.

Definition

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in (\mathbb{F}_q)^n$ s.t. $\forall i \neq j,\ x_i \neq x_j$. Then the **Reed–Solomon code** is defined as

$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\mathsf{def}}{=} \{\mathrm{ev}(f)(\boldsymbol{x}) = (f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

Reed–Solomon codes

Let $\mathbb{F}_q[X]_{<k}$ be the set of univariate polynomials with coefficients in $\mathbb{F}_q$ and degree $< k$.

Definition

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in (\mathbb{F}_q)^n$ s.t. $\forall i \neq j,\ x_i \neq x_j$. Then the **Reed–Solomon code** is defined as

$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\text{def}}{=} \{\mathrm{ev}(f)(\boldsymbol{x}) = (f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

The length n is $\leq q$: we can choose up to q distinct elements in $\mathbb{F}_q$.

The dimension is k: a basis of $\mathbb{F}_q[X]_{<k}$ is given by $\{1, X, \ldots, X^{k-1}\}$.

The minimum distance is $n - k + 1$:

- a polynomial f of degree $k-1$ has at most $k-1$ zeros

$$\omega(f) = \#\{f(x_i) \neq 0\} = n - \#\{f(x_i) = 0\} \geq n - (k-1),$$

- the Singleton bound ensures that $d \leq n - k + 1$.

Reed–Solomon codes

Let $\mathbb{F}_q[X]_{<k}$ be the set of univariate polynomials with coefficients in $\mathbb{F}_q$ and degree $< k$.

Definition

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in (\mathbb{F}_q)^n$ s.t. $\forall i \neq j,\ x_i \neq x_j$. Then the **Reed–Solomon code** is defined as

$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\text{def}}{=} \{\mathrm{ev}(f)(\boldsymbol{x}) = (f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

The length n is $\leq q$: we can choose up to q distinct elements in $\mathbb{F}_q$.

The dimension is k: a basis of $\mathbb{F}_q[X]_{<k}$ is given by $\{1, X, \ldots, X^{k-1}\}$.

The minimum distance is $n - k + 1$:

- a polynomial f of degree $k-1$ has at most $k-1$ zeros

$$\omega(f) = \#\{f(x_i) \neq 0\} = n - \#\{f(x_i) = 0\} \geq n - (k-1),$$

- the Singleton bound ensures that $d \leq n - k + 1$.

Reed–Solomon codes have optimal parameters, attaining the Singleton bound

Reed–Solomon codes

Let $\mathbb{F}_q[X]_{<k}$ be the set of univariate polynomials with coefficients in $\mathbb{F}_q$ and degree $< k$.

Definition

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in (\mathbb{F}_q)^n$ s.t. $\forall i \neq j,\ x_i \neq x_j$. Then the **Reed–Solomon code** is defined as

$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\text{def}}{=} \{\mathrm{ev}(f)(\boldsymbol{x}) = (f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

⚠ **The length n is $\leq q \rightsquigarrow$** to construct long Reed-Solomon codes we need big finite fields
(the bigger the q, the less efficient the arithmetic.)

The dimension is k: a basis of $\mathbb{F}_q[X]_{<k}$ is given by $\{1, X, \ldots, X^{k-1}\}$.

The minimum distance is $n - k + 1$:

- a polynomial f of degree $k-1$ has at most $k-1$ zeros

$$\omega(f) = \#\{f(x_i) \neq 0\} = n - \#\{f(x_i) = 0\} \geq n - (k-1),$$

- the Singleton bound ensures that $d \leq n - k + 1$.

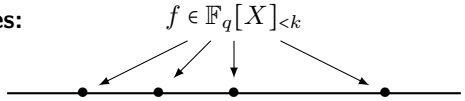
Reed–Solomon codes have optimal parameters, attaining the Singleton bound

1 Linear codes and Reed–Solomon codes

2 Algebraic geometry codes

Algebraic geometry codes (AG codes)

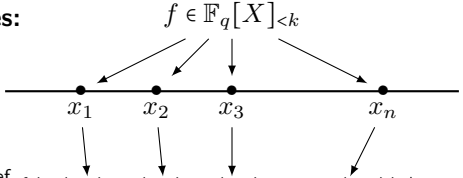
Reed–Solomon (RS) codes:



$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\mathsf{def}}{=} \{(f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

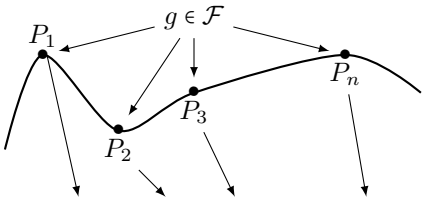
Algebraic geometry codes (AG codes)

Reed–Solomon (RS) codes:



$$\mathsf{RS}_k(\boldsymbol{x}) \stackrel{\text{def}}{=} \{(f(x_1), f(x_2), f(x_3), \ldots, f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$$

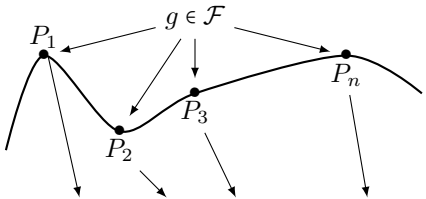
Algebraic Geometry (AG) codes: let $\mathcal{P} = (P_1, \ldots, P_n)$ be a n–tuple of **points** on an **algebraic curve** $\mathcal{X}$ and let $\mathcal{F}$ be a vector space of **functions over the curve**.



$$\mathcal{C}(\mathcal{X}, \mathcal{P}, \mathcal{F}) \stackrel{\text{def}}{=} \{(g(P_1), g(P_2), g(P_3), \ldots, g(P_n)) \mid g \in \mathcal{F}\}$$

Algebraic geometry codes (AG codes)

Algebraic Geometry (AG) codes: let $\mathcal{P} = (P_1, \ldots, P_n)$ be a n–tuple of **points** on an **algebraic curve** $\mathcal{X}$ and let $\mathcal{F}$ be a vector space of **functions over the curve**.



$$\mathcal{C}(\mathcal{X}, \mathcal{P}, \mathcal{F}) \stackrel{\text{def}}{=} \{(g(P_1), g(P_2), g(P_3), \ldots, g(P_n)) \mid g \in \mathcal{F}\}$$

1981: Goppa introduced AG codes from algebraic curves. (also called geometric Goppa codes)

1982: Tsfasman, Vlăduț and Zink designed AG codes above Gilbert–Varshamov bound.

XXs: Various families of curves are studied to get good AG codes.

XXIs: AG codes are used in applications in information theory.

Plane algebraic curves and their functions

1. **Curves and their points:**
   A plane curve over $\mathbb{F}_q$ is defined as the zero set of a bivariate polynomial $f \in \mathbb{F}_q[x, y]$:

$$\mathcal{X} \stackrel{\text{def}}{=} \{(a, b) \in \overline{\mathbb{F}_q}^2 \mid f(a, b) = 0\}.$$

   The *rational* points (or $\mathbb{F}_q$–points) are the points with coordinates lying in $\mathbb{F}_q$.
   The set of $\mathbb{F}_q$–points of the curve $\mathcal{X}$ is denoted by $\mathcal{X}(\mathbb{F}_q) \stackrel{\text{def}}{=} \{(a, b) \in \mathbb{F}_q{}^2 \mid f(a, b) = 0\}$.

Plane algebraic curves and their functions

1. **Curves and their points:**
   A plane curve over $\mathbb{F}_q$ is defined as the zero set of a bivariate polynomial $f \in \mathbb{F}_q[x, y]$:

$$\mathcal{X} \stackrel{\text{def}}{=} \{(a, b) \in \overline{\mathbb{F}_q}^2 \mid f(a, b) = 0\}.$$

   The *rational* points (or $\mathbb{F}_q$–points) are the points with coordinates lying in $\mathbb{F}_q$.
   The set of $\mathbb{F}_q$–points of the curve $\mathcal{X}$ is denoted by $\mathcal{X}(\mathbb{F}_q) \stackrel{\text{def}}{=} \{(a, b) \in \mathbb{F}_q{}^2 \mid f(a, b) = 0\}$.

2. **Functions over a plane curve:**

Plane algebraic curves and their functions

1. **Curves and their points:**
   A plane curve over $\mathbb{F}_q$ is defined as the zero set of a bivariate polynomial $f \in \mathbb{F}_q[x, y]$:

$$\mathcal{X} \stackrel{\text{def}}{=} \{(a, b) \in \overline{\mathbb{F}_q}^2 \mid f(a, b) = 0\}.$$

   The *rational* points (or $\mathbb{F}_q$–points) are the points with coordinates lying in $\mathbb{F}_q$.
   The set of $\mathbb{F}_q$–points of the curve $\mathcal{X}$ is denoted by $\mathcal{X}(\mathbb{F}_q) \stackrel{\text{def}}{=} \{(a, b) \in \mathbb{F}_q{}^2 \mid f(a, b) = 0\}$.

2. **Functions over a plane curve:**

   The *function field* $\mathbb{F}_q(\mathcal{X})$ of a plane curve $\mathcal{X}$ defined by $f = 0$ is

$$\begin{aligned}\mathbb{F}_q(\mathcal{X}) &\stackrel{\text{def}}{=} \text{Frac}\,(\mathbb{F}_q[x, y]/\langle f \rangle) \\ &= \left\{ \frac{h_1}{h_2} : h_1,\, h_2 \in \mathbb{F}_q[x, y] \text{ s.t. } f \nmid h_2 \right\} / \sim \text{ where } \frac{h_1}{h_2} \sim \frac{h_1'}{h_2'} \text{ iff } f \mid h_1 h_2' - h_1' h_2.\end{aligned}$$

Divisors on a curve and Riemann–Roch spaces

Definition

A divisor on a curve $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$ in which the coefficients $n_P \in \mathbb{Z}$ are almost all zero. The support of D is the finite set $\operatorname{Supp} D \stackrel{\text{def}}{=} \{P \in \mathcal{X} \mid n_p \neq 0\}$.

Divisors on a curve and Riemann–Roch spaces

Definition

A divisor on a curve $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$ in which the coefficients $n_P \in \mathbb{Z}$ are almost all zero. The support of D is the finite set $\operatorname{Supp} D \stackrel{\text{def}}{=} \{P \in \mathcal{X} \mid n_p \neq 0\}$.

The set of divisors on $\mathcal{X}$ is endowed with a partial *order*: $D \leq D'$ if $n_P \leq n'_P$ for every point P.

Divisors on a curve and Riemann–Roch spaces

Definition

A divisor on a curve $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$ in which the coefficients $n_P \in \mathbb{Z}$ are almost all zero. The support of D is the finite set $\operatorname{Supp} D \stackrel{\text{def}}{=} \{P \in \mathcal{X} \mid n_p \neq 0\}$.

The set of divisors on $\mathcal{X}$ is endowed with a partial *order*: $D \leq D'$ if $n_P \leq n'_P$ for every point P.
Any non–zero function $g = h_1/h_2$ on $\mathcal{X}$ defines a divisor

$$\operatorname{div}(g) = \sum_{P \in \mathcal{X}} v_P(g) P,$$

where $v_P(g)$ is the *valuation* of g at P ($v_P(g) > 0$ if P is a zero of h_1, $v_P(g) < 0$ if P is a zero of h_2)

Divisors on a curve and Riemann–Roch spaces

Definition

A divisor on a curve $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$ in which the coefficients $n_P \in \mathbb{Z}$ are almost all zero. The support of D is the finite set $\operatorname{Supp} D \stackrel{\text{def}}{=} \{P \in \mathcal{X} \mid n_p \neq 0\}$.

The set of divisors on $\mathcal{X}$ is endowed with a partial *order*: $D \leq D'$ if $n_P \leq n'_P$ for every point P.
Any non–zero function $g = h_1/h_2$ on $\mathcal{X}$ defines a divisor

$$\operatorname{div}(g) = \sum_{P \in \mathcal{X}} v_P(g) P,$$

where $v_P(g)$ is the *valuation* of g at P ($v_P(g) > 0$ if P is a zero of h_1, $v_P(g) < 0$ if P is a zero of h_2)

3. The **Riemann–Roch space** associated to a divisor $D = \sum n_P P$ is the $\mathbb{F}_q$-vector space
$$L(D) = \{g = h_1/h_2 \in \mathbb{F}_q(\mathcal{X}) \mid D \geq -\operatorname{div}(g)\}.$$

- if $n_P < 0$ then P must be a zero of h_1 (of multiplicity $\geqslant -n_P$),
- if $n_P > 0$ then P can be a zero of h_2 (of multiplicity $\leqslant n_P$),
- h_2 has no other zeros outside the points P with $n_P > 0$.

Divisors on a curve and Riemann–Roch spaces

Definition

A divisor on a curve $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$ in which the coefficients $n_P \in \mathbb{Z}$ are almost all zero. The support of D is the finite set $\operatorname{Supp} D \stackrel{\text{def}}{=} \{P \in \mathcal{X} \mid n_p \neq 0\}$.

The set of divisors on $\mathcal{X}$ is endowed with a partial *order*: $D \leq D'$ if $n_P \leq n'_P$ for every point P.
Any non–zero function $g = h_1/h_2$ on $\mathcal{X}$ defines a divisor

$$\operatorname{div}(g) = \sum_{P \in \mathcal{X}} v_P(g) P,$$

where $v_P(g)$ is the *valuation* of g at P ($v_P(g) > 0$ if P is a zero of h_1, $v_P(g) < 0$ if P is a zero of h_2)

3. The **Riemann–Roch space** associated to a divisor $D = \sum n_P P$ is the $\mathbb{F}_q$-vector space
$$L(D) = \{g = h_1/h_2 \in \mathbb{F}_q(\mathcal{X}) \mid D \geq -\operatorname{div}(g)\}.$$

- if $n_P < 0$ then P must be a zero of h_1 (of multiplicity $\geqslant -n_P$),
- if $n_P > 0$ then P can be a zero of h_2 (of multiplicity $\leqslant n_P$),
- h_2 has no other zeros outside the points P with $n_P > 0$.

Computing a basis of $L(D)$ on any $\mathcal{X}$ is hard!

Fix two points $P, Q \in \mathcal{X}(\mathbb{F}_q)$. Then

$$L(mP) = \{g = h_1/h_2 \in \mathbb{F}_q(\mathcal{X}) \mid h_2 \text{ has a zero of order at most } m \text{ at } P\},$$
$$L(mP - nQ) = \{g = h_1/h_2 \in L(mP) \mid h_1 \text{ vanishes with order at least } n \text{ at } Q\}.$$

Fix two points $P, Q \in \mathcal{X}(\mathbb{F}_q)$. Then

$$L(mP) = \{g = h_1/h_2 \in \mathbb{F}_q(\mathcal{X}) \mid h_2 \text{ has a zero of order at most } m \text{ at } P\},$$
$$L(mP - nQ) = \{g = h_1/h_2 \in L(mP) \mid h_1 \text{ vanishes with order at least } n \text{ at } Q\}.$$

Some MAGMA code :

```
> K:=FiniteField(11);
> R<x>:=PolynomialRing(K);
> E:=EllipticCurve(x^3+x);                    defines the curve y^2 = x^3 + x
> P:=PointsAtInfinity(E)[1];                   (unique) point at infinity
> FF<x,y>:=FunctionField(E);
> Basis(5*Divisor(P));     return a basis of the Riemann-Roch space L(5P_∞)
[x*y,y,x^2,x,1]
> Basis(5*Divisor(P)-Divisor(E ! [0,0,1]));          basis of L(5P_∞ − P_0)
[x*y,y,x^2,x]
```

AG codes

Let $\mathcal{X}$ be a curve defined over $\mathbb{F}_q$, a divisor D on $\mathcal{X}$ and $\mathcal{P} = \{P_1, \ldots, P_n\} \subseteq \mathcal{X}(\mathbb{F}_q)$ such that $\mathcal{P} \cap \operatorname{Supp} D = \varnothing$. We define the associated Algebraic Geometry code (or AG code) as

$$\mathcal{C}(\mathcal{X}, \mathcal{P}, D) \stackrel{\text{def}}{=} \{\operatorname{ev}_{\mathcal{P}}(h) = (h(P_1), \ldots, h(P_n)) \mid h \in L(D)\}.$$

If $P \in \mathcal{P}$ with $n_P > 0$, functions in $L(D)$ may have poles at P and the evaluation is not well defined. If $n_P < 0$, the coordinate corresponding to P is always zero.

Parameters of AG codes

An algebraic curve $\mathcal{X}$ comes with a geometric invariant, its genus $g \in \mathbb{N}$.
The genus of a *plane curve* defined by a degree m polynomial is equal to $g = \frac{(m-1)(m-2)}{2}$.

Length $n = \#\mathcal{P} \leq \#\mathcal{X}(\mathbb{F}_q)$.

Hasse–Weil–Serre bound

The number of $\mathbb{F}_q$–points of a smooth projective curve $\mathcal{X}$ defined over $\mathbb{F}_q$ satisfies

$$\#\mathcal{X}(\mathbb{F}_q) \leq q + 1 + 2g\sqrt{q}.$$

Parameters of AG codes

An algebraic curve $\mathcal{X}$ comes with a geometric invariant, its genus $g \in \mathbb{N}$.
The genus of a *plane curve* defined by a degree m polynomial is equal to $g = \frac{(m-1)(m-2)}{2}$.

Length $n = \#\mathcal{P} \leq \#\mathcal{X}(\mathbb{F}_q)$.

Hasse–Weil–Serre bound

The number of $\mathbb{F}_q$–points of a smooth projective curve $\mathcal{X}$ defined over $\mathbb{F}_q$ satisfies

$$\#\mathcal{X}(\mathbb{F}_q) \leq q + 1 + 2g\sqrt{q}.$$

Dimension $k \leq$ dimension of $L(D)$.

Riemann–Roch Theorem

Let $D = \sum n_p P$ such that $\operatorname{Supp} D \subseteq \mathcal{X}(\mathbb{F}_q)$. Define $\deg D \stackrel{\text{def}}{=} \sum n_P$. Then

$$\dim L(D) \geq \deg D + 1 - g, \text{ with equality if } \deg D \geq 2g - 1.$$

Parameters of AG codes

An algebraic curve $\mathcal{X}$ comes with a geometric invariant, its genus $g \in \mathbb{N}$.
The genus of a *plane curve* defined by a degree m polynomial is equal to $g = \frac{(m-1)(m-2)}{2}$.

Length $n = \#\mathcal{P} \leq \#\mathcal{X}(\mathbb{F}_q)$.

Hasse–Weil–Serre bound

The number of $\mathbb{F}_q$–points of a smooth projective curve $\mathcal{X}$ defined over $\mathbb{F}_q$ satisfies

$$\#\mathcal{X}(\mathbb{F}_q) \leq q + 1 + 2g\sqrt{q}.$$

Dimension $k \leq$ dimension of $L(D)$.

Riemann–Roch Theorem

Let $D = \sum n_p P$ such that $\operatorname{Supp} D \subseteq \mathcal{X}(\mathbb{F}_q)$. Define $\deg D \stackrel{\text{def}}{=} \sum n_P$. Then

$$\dim L(D) \geq \deg D + 1 - g, \text{ with equality if } \deg D \geq 2g - 1.$$

Minimum distance $d \geq d^*$ where $d^* \stackrel{\text{def}}{=} n - \deg D$.

$\Rightarrow$ If $2g - 1 \leq \deg(D) < n$, then $\dim(C(\mathcal{X}, \mathcal{P}, D)) = \deg D - g + 1$.

$\Rightarrow n + 1 - g \leq k + d \leq n + 1$. $\rightsquigarrow$ AG codes are g-far from optimality.

Encoding and decoding of AG codes

To use an AG code $C(\mathcal{X}, \mathcal{P}, D)$ in practice we need to

1. **encode:**

2. **decode:**

Encoding and decoding of AG codes

To use an AG code $C(\mathcal{X}, \mathcal{P}, D)$ in practice we need to

1. **encode:** basis of $L(D)$ + (fast) evaluation at points of $\mathcal{P}$;

   Several algorithms to compute Riemann–Roch spaces:
   - Arithmetic method
     Hensel–Landberg (1902), Coated (1970), Davenport (1981), Hess (2001)...
   - Geometric method
     Goppa, Le Brigand–Risler (80's), Huang–Ierardi (90's), Khuri–Makdisi (2007), Le Gluher–Spaenlehauer (2018), Abelard–B–Couvreur–Lecerf (2022),...

   Fast encoding on families of curves with structured $\mathcal{P}$ Beelen–Rosenkilde–Solomatov (2020)

2. **decode:**

Encoding and decoding of AG codes

To use an AG code $C(\mathcal{X}, \mathcal{P}, D)$ in practice we need to

1. **encode:** basis of $L(D)$ + (fast) evaluation at points of $\mathcal{P}$;

   Several algorithms to compute Riemann–Roch spaces:
   - Arithmetic method
     Hensel–Landberg (1902), Coated (1970), Davenport (1981), Hess (2001)...
   - Geometric method
     Goppa, Le Brigand–Risler (80's), Huang–Ierardi (90's), Khuri–Makdisi (2007), Le Gluher–Spaenlehauer (2018), Abelard–B–Couvreur–Lecerf (2022),...

   Fast encoding on families of curves with structured $\mathcal{P}$ Beelen–Rosenkilde–Solomatov (2020)

2. **decode:**

   - Unique decoding — Pelikaan (1992), Kötter (1992)
   - List decoding — Couvreur–Panaccione (2020)

Encoding and decoding of AG codes

To use an AG code $C(\mathcal{X}, \mathcal{P}, D)$ in practice we need to

1. **encode:** basis of $L(D)$ + (fast) evaluation at points of $\mathcal{P}$;

   Several algorithms to compute Riemann–Roch spaces:
   - Arithmetic method
     Hensel–Landberg (1902), Coated (1970), Davenport (1981), Hess (2001)...
   - Geometric method
     Goppa, Le Brigand–Risler (80's), Huang–Ierardi (90's), Khuri–Makdisi (2007), Le Gluher–Spaenlehauer (2018), Abelard–B–Couvreur–Lecerf (2022),...

   Fast encoding on families of curves with structured $\mathcal{P}$ Beelen–Rosenkilde–Solomatov (2020)

2. **decode:**

   - Unique decoding — Pelikaan (1992), Kötter (1992)
   - List decoding — Couvreur–Panaccione (2020)

Thank you for your attention!