

Algebraic-geometry codes in SINGULAR

Stanislav Bulygin

S³CM: Soria Summer School on Computational Mathematics
"Algebraic Coding Theory"

July 3, 2008

Outline of the lecture

- Quick introduction to algebraic function fields.
- Very basics of AG-codes.
- Basic decoding algorithm.

Quick introduction to algebraic function fields

Agenda

We illustrate the most important notions connected with algebraic function fields (FF) with an example of rational function field.

Notions we need for AG-codes construction are

- Valuation ring, valuation (order function).
- Place, residue class field, and residue class map.
- Divisor, principal divisor; degree and dimension of a divisor.

Throughout this part K denotes arbitrary field.

Quick introduction to algebraic function fields

Rational function field

Consider $K(x)$ the rational function field in one variable x over K . For a monic irreducible polynomial $p(x) \in K[x]$ consider the *valuation ring* $\mathcal{O}_{p(x)}$ and its *place* $P_{p(x)}$

$$\mathcal{O}_{p(x)} := \left\{ \frac{f(x)}{g(x)} \mid f(x), g(x) \in K[x], p(x) \nmid g(x) \right\}$$

$$P_{p(x)} = \left\{ \frac{f(x)}{g(x)} \mid f(x), g(x) \in K[x], p(x) \mid f(x), p(x) \nmid g(x) \right\}.$$

We write $P_\alpha := P_{x-\alpha}$ for $\alpha \in K$. Consider also

$$\mathcal{O}_\infty := \left\{ \frac{f(x)}{g(x)} \mid f(x), g(x) \in K[x], \deg(f(x)) \leq \deg(g(x)) \right\}$$

with the place

$$P_\infty = \left\{ \frac{f(x)}{g(x)} \mid f(x), g(x) \in K[x], \deg(f(x)) < \deg(g(x)) \right\}$$

Quick introduction to algebraic function fields

Algebraic function field

An *algebraic function field* F/K (of one variable) over K is an extension of K such that F is a finite algebraic extension of $K(x)$.

Valuation ring and place

Note that for $\mathcal{O}_{P(x)}$ holds that for every $f \in K(x) \setminus \{0\}$ either $f \in \mathcal{O}_{P(x)}$ or $f^{-1} \in \mathcal{O}_{P(x)}$. The same holds for \mathcal{O}_{∞} . This property together with $K \subsetneq \mathcal{O} \subsetneq F$ makes a ring \mathcal{O} the *valuation ring*. The valuation ring \mathcal{O} has a unique maximal ideal, e.g. for $\mathcal{O}_{P(x)}$ it is $P_{P(x)}$, and for \mathcal{O}_{∞} it is P_{∞} . Such a maximal ideal is called the *place* of \mathcal{O} .

Quick introduction to algebraic function fields

Algebraic function field

An *algebraic function field* F/K (of one variable) over K is an extension of K such that F is a finite algebraic extension of $K(x)$.

Valuation ring and place

Note that for $\mathcal{O}_{p(x)}$ holds that for every $f \in K(x) \setminus \{0\}$ either $f \in \mathcal{O}_{p(x)}$ or $f^{-1} \in \mathcal{O}_{p(x)}$. The same holds for \mathcal{O}_∞ . This property together with $K \subsetneq \mathcal{O} \subsetneq F$ makes a ring \mathcal{O} the *valuation ring*. The valuation ring \mathcal{O} has a unique maximal ideal, e.g. for $\mathcal{O}_{p(x)}$ it is $P_{p(x)}$, and for \mathcal{O}_∞ it is P_∞ . Such a maximal ideal is called the *place of \mathcal{O}* .

Quick introduction to algebraic function fields

Rational function field cont.

Consider again $\mathcal{O}_{p(x)}$ and $P_{p(x)}$. Every $h \in K(x) \setminus \{0\}$ can be written in the form $h = p(x)^n \cdot (f(x)/g(x))$ with $n \in \mathbb{Z}$, $f(x), g(x) \in K[x]$, $p(x) \nmid f(x)$, $p(x) \nmid g(x)$. The *valuation* (or order) of h at $P = P_{p(x)}$ is $v_P(h) := n$. Next $K(x)_P = \mathcal{O}_P/P$ is a field since P is maximal. This field is called *residue class field* of P . In this case $K(x)_P \cong K[x]/(p(x))$.

Going general again

A *discrete valuation* of F/K is a function $v : F \rightarrow \mathbb{Z} \cup \{\infty\}$ that generalizes the idea of the order above. For a place P it holds that $v_P(h) \geq 0 \iff h \in \mathcal{O}_P$ and $v_P(h) > 0 \iff h \in P$.

Quick introduction to algebraic function fields

Rational function field cont.

Consider again $\mathcal{O}_{p(x)}$ and $P_{p(x)}$. Every $h \in K(x) \setminus \{0\}$ can be written in the form $h = p(x)^n \cdot (f(x)/g(x))$ with $n \in \mathbb{Z}$, $f(x), g(x) \in K[x]$, $p(x) \nmid f(x)$, $p(x) \nmid g(x)$. The *valuation* (or order) of h at $P = P_{p(x)}$ is $v_P(h) := n$. Next $K(x)_P = \mathcal{O}_P/P$ is a field since P is maximal. This field is called *residue class field* of P . In this case $K(x)_P \cong K[x]/(p(x))$.

Going general again

A *discrete valuation* of F/K is a function $v : F \rightarrow \mathbb{Z} \cup \{\infty\}$ that generalizes the idea of the order above. For a place P it holds that $v_P(h) \geq 0 \iff h \in \mathcal{O}_P$ and $v_P(h) > 0 \iff h \in P$.

Quick introduction to algebraic function fields

Residue class field and map

It can be shown that actually K can be embedded in the residue class field F_P for any place P . Thus F_P can be seen as an extension of K . This extension is finite and $\deg P := [F_P : K]$ is called the *degree* of P . A *residue class map* $\phi : F \rightarrow F_P \cup \{\infty\}$ maps functions from \mathcal{O}_P to their classes modulo P (for a function f it is denoted as $f(P)$) and other functions from F to infinity.

Degree one places in rational FF

For the case $P = P_\alpha$ we have that $K(x)_P \cong K$, so $\deg P_\alpha = 1$ and residue class map restricted to \mathcal{O}_P is simply an evaluation at α . Similarly $\deg P_\infty = 1$ and the valuation is defined as $v_{P_\infty}(h) := \deg g(x) - \deg f(x)$, where $h(x) = f(x)/g(x)$.

Question: What is the residue class map of P_∞ ?

Quick introduction to algebraic function fields

Residue class field and map

It can be shown that actually K can be embedded in the residue class field F_P for any place P . Thus F_P can be seen as an extension of K . This extension is finite and $\deg P := [F_P : K]$ is called the *degree* of P . A *residue class map* $\phi : F \rightarrow F_P \cup \{\infty\}$ maps functions from \mathcal{O}_P to their classes modulo P (for a function f it is denoted as $f(P)$) and other functions from F to infinity.

Degree one places in rational FF

For the case $P = P_\alpha$ we have that $K(x)_P \cong K$, so $\deg P_\alpha = 1$ and residue class map restricted to \mathcal{O}_P is simply an evaluation at α .

Similarly $\deg P_\infty = 1$ and the valuation is defined as $v_{P_\infty}(h) := \deg g(x) - \deg f(x)$, where $h(x) = f(x)/g(x)$.

Question: What is the residue class map of P_∞ ?

Quick introduction to algebraic function fields

Divisors

A *divisor* is a formal sum of places with integer coefficients, i.e. D is a divisor when $D = \sum_{P \in \mathbb{P}_F} n_P P$, where almost all $n_P = 0$; here \mathbb{P}_F is the set of places of F/K . Some technical stuff:

- $\text{supp}(D) := \{P \in \mathbb{P}_F \mid n_P \neq 0\}$.
- $D + D' = \sum_{P \in \mathbb{P}_F} (n_P + n'_P) P$.
- If $D = \sum n_P P \in \mathcal{D}_F$ we denote $v_Q(D) := n_Q$.
- $D_1 \leq D_2 \iff v_P(D_1) \leq v_P(D_2) \forall P \in \mathbb{P}_F$.

The degree of a divisor is $\deg D := \sum_{P \in \mathbb{P}_F} v_P(D) \deg P$.

Principal divisor

$(x)_0 := \sum_{P \in Z} v_P(x) P$ is a *zero divisor* of x , where $Z \subset \mathbb{P}_F$ is a set of zeroes of x (places P with $v_P(x) > 0$). $(x)_\infty := \sum_{P \in N} v_P(x) P$ is a *pole divisor* of x , where $N \subset \mathbb{P}_F$ is a set of poles of x (places P with $v_P(x) < 0$). $(x) = (x)_0 - (x)_\infty$ is a *principal divisor* of x . It is known that $\deg(x) = 0$.

Quick introduction to algebraic function fields

Divisors

A *divisor* is a formal sum of places with integer coefficients, i.e. D is a divisor when $D = \sum_{P \in \mathbb{P}_F} n_P P$, where almost all $n_P = 0$; here \mathbb{P}_F is the set of places of F/K . Some technical stuff:

- $\text{supp}(D) := \{P \in \mathbb{P}_F \mid n_P \neq 0\}$.
- $D + D' = \sum_{P \in \mathbb{P}_F} (n_P + n'_P) P$.
- If $D = \sum n_P P \in \mathcal{D}_F$ we denote $v_Q(D) := n_Q$.
- $D_1 \leq D_2 \iff v_P(D_1) \leq v_P(D_2) \forall P \in \mathbb{P}_F$.

The degree of a divisor is $\deg D := \sum_{P \in \mathbb{P}_F} v_P(D) \deg P$.

Principal divisor

$(x)_0 := \sum_{P \in Z} v_P(x) P$ is a *zero divisor* of x , where $Z \subset \mathbb{P}_F$ is a set of zeroes of x (places P with $v_P(x) > 0$). $(x)_\infty := \sum_{P \in N} v_P(x) P$ is a *pole divisor* of x , where $N \subset \mathbb{P}_F$ is a set of poles of x (places P with $v_P(x) < 0$). $(x) = (x)_0 - (x)_\infty$ is a *principal divisor* of x . It is known that $\deg(x) = 0$.

Quick introduction to algebraic function fields

Dimension of a divisor

For a divisor D the following set is defined

$\mathcal{L}(D) := \{f \in F \mid (f) \geq -D\} \cup \{0\}$. This set is a vector space over K of finite dimension. We denote $\dim D = \dim \mathcal{L}(D)$.

Genus

A *genus* is one of the most important invariants of a function field. It is defined as $g := \max\{\deg D - \dim D + 1\}$, where \max runs over all divisors D .

Quick introduction to algebraic function fields

Dimension of a divisor

For a divisor D the following set is defined

$\mathcal{L}(D) := \{f \in F \mid (f) \geq -D\} \cup \{0\}$. This set is a vector space over K of finite dimension. We denote $\dim D = \dim \mathcal{L}(D)$.

Genus

A *genus* is one of the most important invariants of a function field. It is defined as $g := \max\{\deg D - \dim D + 1\}$, where \max runs over all divisors D .

Very basics of AG-codes

Motivation

Why are algebraic-geometry (AG) codes interesting?

- Have good parameters;
- Outstanding asymptotic performance;
- Effective decoding algorithms are available.

The AG-codes were introduced by V.D.Goppa in early 1980's.

Very basics of AG-codes

Definition

Objects we are working with: F/\mathbb{F}_q an algebraic function field of genus g , P_1, \dots, P_n are pairwise distinct places of F/\mathbb{F}_q of degree 1 (rational places), the divisor $D = P_1 + \dots + P_n$, a divisor G of F/\mathbb{F}_q such that $\text{supp}G \cap \text{supp}D = \emptyset$. The *geometric Goppa code* $C_{\mathcal{L}}(D, G)$ associated with the divisors D and G is the following set

$$C_{\mathcal{L}}(D, G) := \{(x(P_1), \dots, x(P_n)) \mid x \in \mathcal{L}(G)\} \subset \mathbb{F}_q^n.$$

Note that since $\text{supp}G \cap \text{supp}D = \emptyset$, $v_{P_i}(x) \geq 0 \forall i$, and since $\deg P_i = 1 \forall i$, we have that $x(P_i) \in \mathbb{F}_q \forall i$.

Very basics of AG-codes

Linearity

$C_{\mathcal{L}}(D, G)$ is in fact the image of the *evaluation map* $ev : \mathcal{L}(G) \rightarrow \mathbb{F}_q^n$ defined by $x \mapsto (x(P_1), \dots, x(P_n))$. It can be seen that ev is linear, so $C_{\mathcal{L}}(D, G)$ is a linear subspace of \mathbb{F}_q^n , thus a linear code.

Parameters I

$C_{\mathcal{L}}(D, G)$ is an $[n, k, d]_q$ -code with the parameters

$$k = \dim(G) - \dim(G - D), d \geq n - \deg(G).$$

Very basics of AG-codes

Linearity

$C_{\mathcal{L}}(D, G)$ is in fact the image of the *evaluation map* $ev : \mathcal{L}(G) \rightarrow \mathbb{F}_q^n$ defined by $x \mapsto (x(P_1), \dots, x(P_n))$. It can be seen that ev is linear, so $C_{\mathcal{L}}(D, G)$ is a linear subspace of \mathbb{F}_q^n , thus a linear code.

Parameters I

$C_{\mathcal{L}}(D, G)$ is an $[n, k, d]_q$ -code with the parameters

$$k = \dim(G) - \dim(G - D), d \geq n - \deg(G).$$

Very basics of AG-codes

Parameters II

Let $\deg(G) < n$. Then the map $ev_D : \mathcal{L}(G) \rightarrow \mathbb{F}_q^n$ is injective, moreover we have:

- 1 $C_{\mathcal{L}}(D, G)$ is an $[n, k, d]$ -code with

$$k = \dim(G) \geq \deg(G) + 1 - g, d \geq n - \deg(G).$$

where g is the genus of F/\mathbb{F}_q .

- 2 Moreover if $2g - 2 < \deg(G) < n$, then $k = \deg(G) + 1 - g$.
- 3 If $\{x_1, \dots, x_k\}$ is a basis of $\mathcal{L}(G)$ then the matrix

$$M = \begin{pmatrix} x_1(P_1) & x_1(P_2) & \dots & x_1(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ x_k(P_1) & x_k(P_2) & \dots & x_k(P_n) \end{pmatrix}$$

is a generator matrix for the code $C_{\mathcal{L}}(D, G)$.

Very basics of AG-codes

Rational geometric code

If $F = \mathbb{F}_q(z)$ is the rational function field of \mathbb{F}_q in one variable, then $C_{\mathcal{L}}(D, G)$ is called *rational geometric code*.

Parameters

For the parameters of the rational code $C_{\mathcal{L}}(D, G)$ we have:

- ① $n \leq q + 1$, since $\mathbb{F}_q(z)$ has exactly $q + 1$ places of degree 1: q places $P_{\alpha}, \alpha \in \mathbb{F}_q$ and one "place at infinity" P_{∞} .
- ② $k = 0 \iff \deg G < 0$ and $k = n \iff \deg G > n - 2$.
- ③ For $0 \leq \deg G \leq n - 2$,

$$k = 1 + \deg G, d = n - \deg G.$$

In particular, C is an MDS code.

Very basics of AG-codes

Rational geometric code

If $F = \mathbb{F}_q(z)$ is the rational function field of \mathbb{F}_q in one variable, then $C_{\mathcal{L}}(D, G)$ is called *rational geometric code*.

Parameters

For the parameters of the rational code $C_{\mathcal{L}}(D, G)$ we have:

- 1 $n \leq q + 1$, since $\mathbb{F}_q(z)$ has exactly $q + 1$ places of degree 1: q places $P_{\alpha}, \alpha \in \mathbb{F}_q$ and one "place at infinity" P_{∞} .
- 2 $k = 0 \iff \deg G < 0$ and $k = n \iff \deg G > n - 2$.
- 3 For $0 \leq \deg G \leq n - 2$,

$$k = 1 + \deg G, d = n - \deg G.$$

In particular, C is an MDS code.

Very basics of AG-codes

Related problems

- Finding the number of rational places of F/\mathbb{F}_q .
- Efficient calculation of these places.
- Finding a basis of $\mathcal{L}(G)$.

Solution

Brill-Noether algorithm for finding a basis of $\mathcal{L}(G)$ in SINGULAR's library `BRNOETH.LIB`.

Very basics of AG-codes

Related problems

- Finding the number of rational places of F/\mathbb{F}_q .
- Efficient calculation of these places.
- Finding a basis of $\mathcal{L}(G)$.

Solution

Brill-Noether algorithm for finding a basis of $\mathcal{L}(G)$ in SINGULAR's library `BRNOETH.LIB`.

Very basics of AG-codes

One-point AG-codes

Let P, P_1, \dots, P_n be $n + 1$ distinct rational places of F/\mathbb{F}_q . In the definition of $C_{\mathcal{L}}(D, G)$ take $G = aP$ for some positive a . Then clearly $\text{supp}G \cap \text{supp}D = \emptyset$. Such a code $C_{\mathcal{L}}(D, G)$ is called *one-point AG-code*. These codes are used in practice, because of the ease of parameter estimation.

Weierstraß semigroup

For a rational place P consider a set $WS(P) = \{i \geq 0 \mid \exists x \in F : (x)_{\infty} = iP\}$. Elements of $WS(P)$ are called *pole numbers* of P . It turns out that $WS(P)$ is closed under addition and $\mathbb{N}_0 \setminus WS(P)$ is a finite set, thus $WS(P)$ is a numerical semigroup. Elements from $\mathbb{N}_0 \setminus WS(P)$ are called *gaps*, there are exactly g of them, where g is the genus of F/\mathbb{F}_q .

Very basics of AG-codes

One-point AG-codes

Let P, P_1, \dots, P_n be $n + 1$ distinct rational places of F/\mathbb{F}_q . In the definition of $C_{\mathcal{L}}(D, G)$ take $G = aP$ for some positive a . Then clearly $\text{supp}G \cap \text{supp}D = \emptyset$. Such a code $C_{\mathcal{L}}(D, G)$ is called *one-point AG-code*. These codes are used in practice, because of the ease of parameter estimation.

Weierstraß semigroup

For a rational place P consider a set $WS(P) = \{i \geq 0 \mid \exists x \in F : (x)_{\infty} = iP\}$. Elements of $WS(P)$ are called *pole numbers* of P . It turns out that $WS(P)$ is closed under addition and $\mathbb{N}_0 \setminus WS(P)$ is a finite set, thus $WS(P)$ is a numerical semigroup. Elements from $\mathbb{N}_0 \setminus WS(P)$ are called *gaps*, there are exactly g of them, where g is the genus of F/\mathbb{F}_q .

Very basics of AG-codes

Parameters of one-point AG-codes via a Weierstraß semigroup

Let $C := C_{\mathcal{L}}(D, G)$ be a one-point AG-code with $G = aP$ for some positive integer a . Then $\dim C = |WS(P) \cap \{0, \dots, a\}|$ and $d \geq n - a$. So the knowledge of the Weierstrass semigroup is crucial for understanding the code C .

Basic decoding algorithm

Setting

We will solve the decoding problem up to some designed capacity for the code $C := C_{\mathcal{L}}(D, G)^{\perp}$. Namely, for a received word $\mathbf{r} = \mathbf{c} + \mathbf{e}$ with $\mathbf{c} \in C$ and \mathbf{e} an error vector with $\text{wt}(\mathbf{e}) \leq$ some value, we would like to find \mathbf{e} and thus \mathbf{c} . Let $\mathbf{e} = (e_1, \dots, e_n)$, denote $E := \{i \mid 1 \leq i \leq n : e_i \neq 0\}$ the set of *error positions*, whereas e_i 's are *error values*.

Basic decoding algorithm

Some notation and technical stuff

For $\mathbf{b} = (b_1, \dots, b_n) \in \mathbb{F}_q^n$ and $f \in \mathcal{L}(G)$ we define the *syndrome* $[\mathbf{b}, f] := \sum_{i=1}^n b_i \cdot f(P_i)$. We have that

$$C = \{\mathbf{b} \in \mathbb{F}_q^n \mid [b, f] = 0 \forall f \in \mathcal{L}(G)\}.$$

Let $t \geq 0$ be an integer and G_1 be a divisor of F/\mathbb{F}_q satisfying the following

$$\begin{cases} \text{supp } G_1 \cap \text{supp } D = \emptyset, \\ \deg G_1 < \deg G - (2g - 2) - t, \\ \dim G_1 > t. \end{cases}$$

If G_1 and t satisfy above then $t \leq (d^* - 1)/2$, where $d^* = \deg G - (2g - 2)$ is the *designed distance*. For $0 \leq t \leq (d^* - 1 - g)/2$ there exists a divisor G_1 such that the above is satisfied. So for maximal $t = (d^* - 1 - g)/2$ we have that $\deg G_1 < g + (\deg G - 3g + 1)/2$.

Basic decoding algorithm

Error-locator function

We will be interested in finding *error-locator function* $f \in \mathcal{L}(G_1) \setminus \{0\}$ such that $f(P_i) = 0 \forall i \in E$. If we are able to find such a function, then error positions are among elements of the set $N(f) := \{i \mid 1 \leq i \leq n : f(P_i) = 0\}$. The next step would be to find error values e_i for $i \in N(f)$ (note that $e_i = 0$ for $i \notin N(f)$).

Preparation

Fix bases

$$\begin{aligned}\langle f_1, \dots, f_l \rangle &= \mathcal{L}(G_1), \\ \langle g_1, \dots, g_k \rangle &= \mathcal{L}(G - G_1), \\ \langle h_1, \dots, h_m \rangle &= \mathcal{L}(G).\end{aligned}$$

Note that choice of these bases does not depend on r and that $f_i g_j \in \mathcal{L}(G)$ for $1 \leq i \leq l$ and $1 \leq j \leq k$.

Basic decoding algorithm

Error-locator function

We will be interested in finding *error-locator function* $f \in \mathcal{L}(G_1) \setminus \{0\}$ such that $f(P_i) = 0 \forall i \in E$. If we are able to find such a function, then error positions are among elements of the set $N(f) := \{i \mid 1 \leq i \leq n : f(P_i) = 0\}$. The next step would be to find error values e_i for $i \in N(f)$ (note that $e_i = 0$ for $i \notin N(f)$).

Preparation

Fix bases

$$\begin{aligned}\langle f_1, \dots, f_l \rangle &= \mathcal{L}(G_1), \\ \langle g_1, \dots, g_k \rangle &= \mathcal{L}(G - G_1), \\ \langle h_1, \dots, h_m \rangle &= \mathcal{L}(G).\end{aligned}$$

Note that choice of these bases does not depend on \mathbf{r} and that $f_i g_j \in \mathcal{L}(G)$ for $1 \leq i \leq l$ and $1 \leq j \leq k$.

Basic decoding algorithm

Algorithm

- 1 Find a non-trivial solution $(\alpha_1, \dots, \alpha_l)$ of the system

$$\left\{ \sum_{i=1}^l [\mathbf{r}, f_i \mathbf{g}_j] \cdot x_i = 0, j = 1, \dots, k \right\}.$$

and set $f := \sum_{i=1}^l \alpha_i f_i$. If only trivial solution exists return a failure. With our assumptions on t and G_1 such a non-trivial solution exists and f is the error-locator function.

- 2 Determine $N(f)$ by evaluating $f(P_j) = \sum_{i=1}^l \alpha_i f_i(P_j)$ for $j = 1, \dots, n$.

Basic decoding algorithm

Algorithm contd.

- ③ If the system

$$\left\{ \sum_{i \in N(f)} h_j(P_i) \cdot z_i = [\mathbf{r}, h_j], j = 1, \dots, m \right\}$$

has a unique solution $(e_i)_{i \in N(f)}$, set $\mathbf{e} = (e_1, \dots, e_n)$ with $e_i = 0$ for $i \notin N(f)$. If the system is not uniquely solvable return a failure. With our assumption the system above always has a unique solution.

- ④ Check whether $\mathbf{c} := \mathbf{r} - \mathbf{e}$ is in C by computing the syndromes $[\mathbf{c}, h_i], i = 1, \dots, m$ and whether $\text{wt}(\mathbf{e}) \leq t$. If this is so, then decode \mathbf{r} to \mathbf{c} . If no return a failure.

Basic decoding algorithm

Discussion

- Provided that G_1 and t satisfy our assumptions, the algorithm above corrects all errors of weight $\leq t$.
- One can choose the divisor G_1 in such a way that the algorithm above decodes all errors \mathbf{e} of weight

$$\text{wt}(\mathbf{e}) \leq (d^* - 1 - g)/2.$$

More powerful algorithms that correct more errors will be discussed in the second part of this School.

Implementation

This algorithm was introduced by A.N.Skorobogatov and S.G.Vladut in 1990. Implementation in SINGULAR is available.

Basic decoding algorithm

Discussion

- Provided that G_1 and t satisfy our assumptions, the algorithm above corrects all errors of weight $\leq t$.
- One can choose the divisor G_1 in such a way that the algorithm above decodes all errors \mathbf{e} of weight

$$\text{wt}(\mathbf{e}) \leq (d^* - 1 - g)/2.$$

More powerful algorithms that correct more errors will be discussed in the second part of this School.

Implementation

This algorithm was introduced by A.N.Skorobogatov and S.G.Vladut in 1990. Implementation in SINGULAR is available.

That's it! Questions? Remarks?