Algebraic Geometry and Number Theory with MAGMA
Centre Émile Borel, Institute Henri Poincaré
Paris, October 4–8 2004

Constructing Algebraic-Geometric Codes using MAGMA

Markus Grassl

http://iaks-www.ira.uka.de/home/grassl

Institut für Algorithmen und Kognitive Systeme
Fakultät für Informatik, Universität Karlsruhe
Germany

Error-Correcting Codes (I)

- **block codes:**
  set of words of length n over a finite alphabet $\mathcal{A}$
- **linear block codes:**
  - the alphabet $\mathcal{A}$ is a finite field $\mathbb{F}_q$ (or at least a finite ring, e.g. $\mathbb{Z}/4\mathbb{Z}$)
  - the code is a subspace of $\mathbb{F}_q^n$ of dimension k
- **minimum distance:**
  - Hamming distance:

$$d := \mathrm{d}_{\min}(C) = \min\{\mathrm{d}_H(\boldsymbol{v}, \boldsymbol{w}) : \boldsymbol{v}, \boldsymbol{w} \in C \mid \boldsymbol{v} \neq \boldsymbol{w}\}$$

$$\text{where } \mathrm{d}_H(\boldsymbol{v}, \boldsymbol{w}) := \left|\{i : i \in \{1, \ldots, n\} \mid v_i \neq w_i\}\right|$$

  - for linear codes:

$$d = \min\{\mathrm{wgt}(\boldsymbol{c}) : \boldsymbol{c} \in C \setminus \{\boldsymbol{0}\}\}$$

- notation:

$$C = [n, k, d]_q$$

Error-Correcting Codes (II)

- error-detection:
  detection of all errors $\boldsymbol{e}$ of weight $\mathrm{wgt}(\boldsymbol{e}) < d$
- error-correction:
  correction of all errors $\boldsymbol{e}$ of weight $\mathrm{wgt}(\boldsymbol{e}) \leq (d-1)/2$

Goal: Find codes $C = [n, k, d]$ with high minimum distance d and high rate $R = k/n$.

Problem: Computing the minimum distance of a random linear binary code is NP-hard [Vardy 1997].

$\Longrightarrow$ Derive good lower bounds on the minimum distance for special classes of codes.

in MAGMA:
database of the best know linear codes over $GF(2)$, $GF(3)$, and $GF(4)$ (so far) based on Brouwer's tables, e.g.

```
> c:=BKLC(GF(2),200,40);
```

AG Codes from Riemann-Roch Spaces

Ingredients:

C	a curve over $\mathbb{F}_q$ with genus g
$P_1, \ldots, P_n$	mutually distinct rational points on C
$P := P_1 + \ldots + P_n$	
D	a divisor on C with $\mathrm{supp}(D) \cap \{P_1, \ldots, P_n\} = \emptyset$
$\mathcal{L}(D)$	Riemann-Roch space

***functional* AG Code:**

$$C_{\mathcal{L}}(P, D) := \big\{(f(P_1), \ldots, f(P_n)) : f \in \mathcal{L}(D)\big\}$$

Parameters:

length	$n \leq$ #rational points on C
dimension	$k = \dim \mathcal{L}(D) - \dim \mathcal{L}(D - P)$
minimum distance	$d \geq n - \deg D$
for $\deg D < n$:	$k = \dim \mathcal{L}(D) \geq \deg D + 1 - g$
	$\Longrightarrow \boxed{k + d \geq n + 1 - g}$

AG Codes from Riemann-Roch Spaces with MAGMA

```
> K<w>:=GF(16);
> P<x,y>:=PolynomialRing(K,2);
>
> f:=x^5+x^4*y+x^4+x^3*y^2+x^3*y+x^2*y^2+x^2+x*y^3+x*y+y^3+y^2;
> C:=Curve(AffineSpace(K,2),f); //genus 4, 45 places
>
> time plcs:=Places(C,1);
Time: 0.581
> SetVerbose("AGCode",true);
> c:=AGCode(plcs[2..#plcs],20*plcs[1]);
Algebraic-geometric code:
    Genus computation time: 0
    Riemann-Roch dimension: 17
    Riemann-Roch space time:  0.03
    Evaluation time: 2.444
Algebraic-geometric code time: 2.474
> c:Minimal;
[44, 17] Linear Code over GF(2^4)
```

AG Codes from Differentials

Ingredients:

C	a curve over $\mathbb{F}_q$ with genus g
$P_1, \ldots, P_n$	mutually distinct rational points on C
$P := P_1 + \ldots + P_n$	
D	a divisor on C with $\mathrm{supp}(D) \cap \{P_1, \ldots, P_n\} = \emptyset$
$\Omega(D)$	space of differentials of D

***differential* AG code:**

$$C_\Omega(P, D) := \left\{ (\omega_{P_1}(1), \ldots, \omega_{P_n}(1)) : \omega \in \Omega(D - P) \right\}$$

Parameters:

length	$n \leq \#$rational points on C
dimension	$k = i(D - P) - i(D)$, where $i(D)$ denotes the *index of speciality*
minimum distance	$d \geq \deg D - (2g - 2)$
for $2g - 2 < \deg D$:	$k = i(D - P) \geq n + g - 1 - \deg D$
	$\Longrightarrow \boxed{k + d \geq n + 1 - g}$

AG Codes from Differentials with MAGMA (I)

```
intrinsic AGCodeOmega(plcs::[PlcCrvElt],D::DivCrvElt)->CodeLinFld
{Return the AG code defined by the divisor D and the places plcs}
  require Curve(D) cmpeq Curve(Universe(plcs)):
    "Arguments must be defined with respect to the same curve";
   // further checking of the arguments

  B:=DifferentialBasis(D-&+plcs); //basis of differentials

  if #B eq 0 then
    return ZeroCode(CoefficientRing(Curve(D)),#plcs);
  end if;

  g:=Matrix([[Residue(w,pl):pl in plcs]:w in B]); //all residues

  return LinearCode(g);

end intrinsic;
```

AG Codes from Differentials with MAGMA (II)

```
> Attach("AGOmega.m");
>
> K<w>:=GF(16);
> P<x,y>:=PolynomialRing(K,2);
> f:=x^5+x^4*y+x^4+x^3*y^2+x^3*y+x^2*y^2+x^2+x*y^3+x*y+y^3+y^2;
>
> C:=Curve(AffineSpace(K,2),f); //genus 4, 45 places
>
> time plcs:=Places(C,1);
Time: 0.601
>
> SetVerbose("AGCode",true);
> c:=AGCodeOmega(plcs[2..#plcs],30*plcs[1]);
Algebraic-geometric code:
  differential space time:  0.181
  evaluation time:          7.631
> c:Minimal;
[44, 17] Linear Code over GF(2^4)
```

AG Codes: Overview

- $C_\Omega(P, D) = C_\mathcal{L}(P, D)^\perp$ (duality of functional/differential AG codes)
- $D_2 \leq D_1 \Longrightarrow C_\mathcal{L}(P, D_2) \leq C_\mathcal{L}(P, D_1)$ (lattice of codes)
- number of rational points on $C =$ max. length of the code
- bounds on the minimum distance

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

Main Goal: Find curves with many rational points (and small genus).

Algorithmic Problems:

- find explicit equation for the curve
- compute all points (and some places of higher degree)
- compute Riemann-Roch spaces/differential spaces
- function evaluation at points/computation of residues

Searching for Good Curves

- Many papers on good curves do not provide enough details (for me) to find explicit equations for the curves.
- In van der Geer's tables, there are several quite general types of the constructions, e.g.

  I. Methods from general class field theory

  V.2 explicit curves, e.g. Hermitean curves, Klein's quartic, Artin-Schreier curves, Kummer extensions, complete intersections or curves obtained by computer search

Consequences:

- Try to search for good curves using MAGMA
  (feasible for random curves with small genus)
- learn more about the theory . . .
- . . . or ask the experts.

Disappointing News . . .

The curve in question was indeed constructed by Chaoping Xing and myself using class field theory. In this particular case things are much simpler since the base field is a rational function field. In particular, the auxiliary field E_n in the construction is just the cyclotomic function field over F_8 modulo the irreducible polynomial $x^2 + x + 1$ over F_8.

Thus, in principle, one should be able to get an explicit description of the desired subfield K_n of E_n, but the computations will be lengthy and can be carried out only with the help of a computer algebra system.

Maybe somebody in the group of Professor Gekeler in Saarbruecken can provide assistance with this since they have done extensive computations with cyclotomic function fields.

Best regards from H. Niederreiter

Some Good News . . .

- With the help of MAX GEBHARDT (Saarbrücken) I was able to construct and verify the curve over $GF(8)$ with genus 9 and 45 rational points.
- Even better, from the Diploma thesis of ALICE KELLER (Saarbrücken), I learned about Carlitz modules.
- Around the same time, ALLAN STEEL implemented polynomial factorisation over function fields in positive characteristics (inseparable case).

Algorithm

- construct the splitting field of a polynomial $f(Y) \in \mathbb{F}_q(x)[Y]$:
  problem: find one particular factor of $f(Y)$, but $f(Y)$ has exponential degree
- construct subfields using the action of the Galois group
  - action of the Galois group can be implemented
  - problem: computation of the minimal polynomial of an element that is fixed by the Galois subgroup

Even Better News

- CLAUS FIEKER implemented ray class fields in MAGMA.
- The efficiency of MAGMA allows to randomly search for good curves.

Algorithm

1. pick some random function field $\mathbb{F}_q(x,y)/F_q(x)$ of small degree
2. pick some random divisor D in that field
3. compute the ray class group R
   `R:=RayClassGroup(D);`
4. pick a random subgroup R_0 of R
5. compute some/all subgroup U with $R_o \leq U \leq R$ of small index
6. compute the genus and number of places of the extension with Galois group R/U
7. if a function field with good parameters is found, compute it explicitly:
   `F:=RayClassField(D,U);`

Results & Outlook

Results

- explicit equations for many curves of van der Geer's tables found
- new curve over $\mathbb{F}_3$ with 20 rational points and genus 10 (upper bound 21)

Open Problems

- resulting equations have high degree/many terms
- singular curves
- computation of the rational points seems to be hard
- evaluation at the rational points seems to be hard

Possible Project:

database of curves with many rational points