ISIT 2000
The IEEE International Symposium
on Information Theory
Sorrento, June 25 –June 30, 2000

On the Minimum Distance of some Quadratic-Residue Codes

Markus Grassl

Arbeitsgruppen
Codierungstheorie & Quantum Computing
Prof. Thomas Beth
Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe, Germany

Outline

- Quotations from MacWilliams & Sloane
- Quadratic residue codes
  - definition
  - properties
- Computing the minimum distance
  - naive method
  - linear programming
  - automorphism group & subcodes
  - lower bounds
- Results & timings

(a) Over GF(2)

n	k	d	n	k	d	n	k	d
8	4	4	74	37	14	138	69	14–22
18	9	6	80	40	16	152	76	20
24	12	8	90	45	18	168	84	16–24
32	16	8	98	49	16	192	96	16–28
42	21	10	104	52	20	194	97	16–28
48	24	12	114	57	12–16	200	100	16–32
72	36	12	128	64	20			

(b) Over GF(3)

n	k	d	n	k	d	n	k	d
12	6	6	48	24	15	74	37	?
14	7	6	60	30	18	84	42	?
24	12	9	62	31	?	98	49	?
38	19	?	72	36	?			

Fig. 16.2. A table of extended quadratic-residue codes $\hat{\mathcal{Q}}$.

The Goal

As the codes in Fig. 16.2 show, d is often greater than the bound given by Theorem 1.

Research Problems (16.1). Fill in the gaps in Fig. 16.2, extend these tables, and compute similar tables for other primes l.

(16.2) How does the minimum distance of QR codes behave as $p \to \infty$?

§3. Idempotents of quadratic-residue codes

The case $l = 2$ is considered first.

Quadratic Residue Codes (I)

- codes of length p (or $p+1$) over $GF(l)$ where p, l prime and $l = a^2 \bmod p$
- $x^p - 1 = (x-1)q(x)n(x) \in GF(l)[x]$ where

$$q(x) = \prod_{r\in Q}(x-\alpha^r) \quad \text{and} \quad n(x) = \prod_{r\in N}(x-\alpha^r)$$

  - Q (resp. N): set of non-zero quadratic residues (non-residues)
  - α: p^{th} root of unity

$\overline{\mathcal{Q}}_p$	d_p+1	cyclic, $(x-1)q(x)$
$\mathcal{Q}_p$	d_p	cyclic, $q(x)$
$\hat{\mathcal{Q}}_p$	d_p+1	extended cyclic

similar family $\mathcal{N}_p$ using $q(x)$ instead of $n(x)$

Quadratic Residue Codes (II)

$p \equiv 1 \pmod 4$:

- $d_p \geq \sqrt{p}$
- $\mathcal{Q}^\perp = \overline{\mathcal{N}}$, $\mathcal{N}^\perp = \overline{\mathcal{Q}}$, $(\hat{\mathcal{Q}})^\perp = \hat{\mathcal{N}}$

$p \equiv 3 \pmod 4$:

- $d_p^2 - d_p + 1 \geq p$
- $\mathcal{Q}^\perp = \overline{\mathcal{Q}}$, $\mathcal{N}^\perp = \overline{\mathcal{N}}$, $\hat{\mathcal{Q}}^\perp = \hat{\mathcal{Q}}$, $\hat{\mathcal{N}}^\perp = \hat{\mathcal{N}}$
- $\hat{\mathcal{Q}}$ is doubly-even $(l = 2)$ or $\forall c \in \hat{\mathcal{Q}} : \mathrm{wgt}(c) \equiv 0 \bmod 3$ $(l = 3)$

$\mathrm{Aut}(\hat{\mathcal{Q}}) \geq PSL(2,p)$:

- $|PSL(2,p)| = \frac{1}{2}(p-1)p(p+1)$
- cyclic subgroups $C_{(p-1)/2}$, C_p, and $C_{(p+1)/2}$

Computing $d_{\min}$

given: a linear code $C = [n, k]_q$

Naive method

enumerate all q^k codewords

$\Longrightarrow$ no lower bounds

does not exploit code structure

NP-hard (A. Vardy, IT-43 (1997))

Linear programming

sample from the q^k codewords

$\Longrightarrow$ additional constraints $A_i \geq a_i$ for LP

may be combined with invariant theory for self-dual codes (cf., e. g., Gleason)

Automorphism group & subcodes

- use the automorphism group in the enumeration/sampling process
- subgroups $H \leq \mathrm{Aut}(C)$: constraints on the weight enumerator:

$$A_i \equiv A_i(H) = \bmod |H|$$

Lower Bounds on $\mathrm{d}_{\mathsf{min}}$

Basic ideas

- systematic encoding

  information: $\boldsymbol{i} = (i_1, i_2, \ldots, i_k)$

  codeword: $\boldsymbol{c} = (i_1, i_2, \ldots, i_k, c_{k+1}, \ldots, c_n)$

  $\Longrightarrow \operatorname{wgt} \boldsymbol{c} \geq \operatorname{wgt} \boldsymbol{i}$

  enumerate vectors $\boldsymbol{i}$ with increasing weight $1, 2, \ldots, r$

- m disjoint information sets

  $\Longrightarrow$ code word of low weight or $\mathrm{d}_{\mathsf{min}} \geq m(r+1)$

- generalization for overlapping information sets (K.-H. Zimmermann)

Estimation of the running time

$$a(l,p,w) := (l-1)^r \binom{\frac{1}{2}(p+1)}{r} \in O\big(\exp(\sqrt{p}\ln p)\big)$$

where $r = \lceil w/2 \rceil - 1$

Updated Table

(a) Over GF(2)

n	k	d
8	4	4
18	9	6
24	12	8
32	16	8
42	21	10
48	24	12
72	36	12

n	k	d
74	37	14
80	40	16
90	45	18
98	49	16
104	52	20
114	57	16[a]
128	64	20

n	k	d
138	69	22[c]
152	76	20
168	84	24[e]
192	96	24[e]–28
194	97	22[f]–28
200	100	24[g]–32

(b) Over GF(3)

n	k	d
12	6	6
14	7	6
24	12	9
38	19	11[b]

n	k	d
48	24	15
60	30	18
62	31	12[a]
72	36	18[a]

n	k	d
74	37	18[a]
84	42	21[d]
98	49	21[h]–24
108	54	21[i]–27

Coppersmith/Seroussi 1984 (lower bounds), Remijn/de Vroedt 1984 (subgroups of $\mathrm{Aut}(C)$)
Boston 1999, Kuhlmann 1999 (MAGMA)
Grassl 2000 (MAGMA plus theory)

Timings

(MAGMA V2.7-1 on an AMD-Athlon-PC under Linux at 800 MHz)

field	n	k	d_{min}	r	time	remark
$GF(2)$	114	57	16	7	92.16 s	—
$GF(2)$	113	57	15	7	46.04 s	cyclic
$GF(2)$	113	56	16	6	5.37 s	cyclic
$GF(2)$	137	68	22	9	10 883.20 s	cyclic
$GF(2)$	167	83	24	10	68 953.97 s	cyclic, doubly even
$GF(3)$	62	31	12	5	2.52 s	—
$GF(3)$	61	30	12	5	1.06 s	cyclic
$GF(3)$	71	35	18	8	1 632.92 s	cyclic
			≥ 17	7	224.60 s	cyclic, $0 \bmod 3$
$GF(3)$	73	36	18	8	2 529.80 s	cyclic
$GF(3)$	83	41	21	9	61 140.65 s	cyclic
			≥ 19	9	8 071.44 s	cyclic, $0 \bmod 3$

$$\overline{\mathcal{Q}} = [113, 56]_2$$

```
Magma V2.7-1      Thu Jun 15 2000 11:49:32 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=ExpurgateCode(QRCode(GF(2),113));
> C:Minimal;
[113, 56] Linear Code over GF(2)
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 113, dimension 56, cyclic
Starting search...
Using congruence d mod 2 = 0
lower: 12, upper: 58, r: 1 time: 0.000000.
new max 38: 0: time 0.000000.
new max 32: 1: time 0.000000.
new max 30: 2: time 0.000000.
new max 26: 10: time 0.000000.
new max 20: 26: time 0.000000.
lower: 12, upper: 20, r: 2 time: 0.000000.
new max 16: 7 48: time 0.000000.
lower: 12, upper: 16, r: 3 time: 0.000000.
lower: 12, upper: 16, r: 4 time: 0.000000.
lower: 12, upper: 16, r: 5 time: 0.050000.
lower: 14, upper: 16, r: 6 time: 0.610000.
Final lower: 16, upper: 16, time: 5.370000.
16
```

$\hat{\mathcal{Q}} = [114, 57]_2$

```
Magma V2.7-1      Thu Jun 15 2000 11:48:18 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=ExtendCode(QRCode(GF(2),113));
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 114, dimension 57,
  not cyclic
Finished constructing sets, time taken: 0.000000.
Ranks: 57 57
Starting search...
Using congruence d mod 2 = 0
r: 1, i: 0, lower: 12, upper: 58 time: 0.000000.
new max 34: 0: time 0.000000.
new max 30: 3: time 0.000000.
new max 26: 4: time 0.000000.
r: 1, i: 1, lower: 12, upper: 26 time: 0.000000.
new max 20: 26: time 0.000000.
r: 2, i: 0, lower: 11, upper: 20 time: 0.000000.
new max 16: 16 38: time 0.000000.
r: 2, i: 1, lower: 11, upper: 16 time: 0.000000.
r: 3, i: 0, lower: 11, upper: 16 time: 0.000000.
r: 3, i: 1, lower: 11, upper: 16 time: 0.000000.
r: 4, i: 0, lower: 11, upper: 16 time: 0.010000.
r: 4, i: 1, lower: 11, upper: 16 time: 0.070000.
r: 5, i: 0, lower: 11, upper: 16 time: 0.130000.
r: 5, i: 1, lower: 11, upper: 16 time: 0.740000.
r: 6, i: 0, lower: 12, upper: 16 time: 1.350000.
r: 6, i: 1, lower: 12, upper: 16 time: 6.750000.
r: 7, i: 0, lower: 14, upper: 16 time: 12.150000.
r: 7, i: 1, lower: 14, upper: 16 time: 52.160000.
Final lower: 16, upper: 16, time: 92.160000.
16
```

$$\mathcal{Q} = [113, 57]_2$$

```
Magma V2.7-1      Thu Jun 15 2000 11:54:58 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=QRCode(GF(2),113);
> C:Minimal;
[113, 57] Quadratic Residue code over GF(2)
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 113, dimension 57, cyclic
Starting search...
lower: 11, upper: 57, r: 1 time: 0.000000.
new max 33: 0: time 0.000000.
new max 30: 3: time 0.000000.
new max 25: 4: time 0.000000.
lower: 11, upper: 25, r: 2 time: 0.000000.
new max 23: 0 14: time 0.000000.
new max 20: 0 27: time 0.000000.
new max 16: 16 38: time 0.000000.
lower: 11, upper: 16, r: 3 time: 0.000000.
new max 15: 18 28 38: time 0.000000.
lower: 11, upper: 15, r: 4 time: 0.010000.
lower: 11, upper: 15, r: 5 time: 0.060000.
lower: 12, upper: 15, r: 6 time: 0.680000.
lower: 14, upper: 15, r: 7 time: 6.070000.
Final lower: 15, upper: 15, time: 46.040000.
15
```

$$\hat{\mathcal{Q}} = [62, 31]_3$$

```
Magma V2.7-1      Thu Jun 15 2000 14:44:28 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=ExtendCode(QRCode(GF(3),61));
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 62, dimension 31, not cyclic
Finished constructing sets, time taken: 0.000000.
Ranks: 31 31
Starting search...
r: 1, i: 0, lower: 8, upper: 32 time: 0.000000.
new max 22: 0 time 0.000000.
new max 18: 2 time 0.000000.
new max 17: 6 time 0.000000.
new max 12: 9 time 0.000000.
r: 1, i: 1, lower: 8, upper: 12 time: 0.000000.
r: 2, i: 0, lower: 8, upper: 12 time: 0.000000.
r: 2, i: 1, lower: 8, upper: 12 time: 0.000000.
r: 3, i: 0, lower: 8, upper: 12 time: 0.000000.
r: 3, i: 1, lower: 8, upper: 12 time: 0.010000.
r: 4, i: 0, lower: 8, upper: 12 time: 0.020000.
r: 4, i: 1, lower: 8, upper: 12 time: 0.120000.
r: 5, i: 0, lower: 10, upper: 12 time: 0.230000.
r: 5, i: 1, lower: 10, upper: 12 time: 1.370000.
Final lower: 12, upper: 12, time: 2.520000.
12
```

$$\boxed{\overline{\mathcal{Q}} = [61,30]_3}$$

```
Magma V2.7-1       Thu Jun 15 2000 11:57:59 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=Dual(QRCode(GF(3),61));
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 61, dimension 30, cyclic
Starting search...
lower: 3, upper: 32, r: 1 time: 0.000000.
new max 18: 0 time 0.000000.
new max 12: 5 time 0.000000.
lower: 5, upper: 12, r: 2 time: 0.000000.
lower: 7, upper: 12, r: 3 time: 0.000000.
lower: 9, upper: 12, r: 4 time: 0.010000.
lower: 11, upper: 12, r: 5 time: 0.100000.
Final lower: 12, upper: 12, time: 1.060000.
12
```

$$\boxed{\overline{\mathcal{Q}} = [71, 35]_3}$$

```
Magma V2.7-1      Thu Jun 15 2000 11:59:53 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=Dual(QRCode(GF(3),71));
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 71, dimension 35, cyclic
Starting search...
lower: 3, upper: 37, r: 1 time: 0.000000.
new max 18: 0 time 0.000000.
lower: 5, upper: 18, r: 2 time: 0.000000.
lower: 7, upper: 18, r: 3 time: 0.000000.
lower: 9, upper: 18, r: 4 time: 0.010000.
lower: 11, upper: 18, r: 5 time: 0.200000.
lower: 13, upper: 18, r: 6 time: 2.540000.
lower: 15, upper: 18, r: 7 time: 26.180000.
lower: 17, upper: 18, r: 8 time: 224.600000.
Final lower: 18, upper: 18, time: 1632.920000.
18
```

$\hat{\mathcal{Q}} = [73, 36]_3$

```
Magma V2.7-1      Thu Jun 15 2000 13:50:05 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> C:=Dual(QRCode(GF(3),73));
> SetVerbose("Code",1);
> MinimumWeight(C);
Code MinimumWeightZimmermann: length 73, dimension 36, cyclic
Starting search...
lower: 3, upper: 38, r: 1 time: 0.000000.
new max 26: 0 time 0.000000.
new max 20: 1 time 0.000000.
lower: 5, upper: 20, r: 2 time: 0.000000.
new max 18: 0, 5 time 0.000000.
lower: 7, upper: 18, r: 3 time: 0.000000.
lower: 9, upper: 18, r: 4 time: 0.020000.
lower: 11, upper: 18, r: 5 time: 0.270000.
lower: 13, upper: 18, r: 6 time: 3.550000.
lower: 15, upper: 18, r: 7 time: 37.900000.
lower: 17, upper: 18, r: 8 time: 336.300000.
Final lower: 18, upper: 18, time: 2529.800000.
18
```

$\hat{\mathcal{Q}} = [83, 41]_3$

```
Magma V2.7-1      Thu Jun 15 2000 14:54:06 on siegbahn
Type ? for help.  Type <Ctrl>-D to quit.
> SetVerbose("Code",1);
> C:=QRCode(GF(3),83);
> D:=Dual(C);
> IsWeaklySelfDual(D);
true
> time MinimumWeight(D);
Code MinimumWeightZimmermann: length 83, dimension 41, cyclic
Starting search...
lower: 3, upper: 43, r: 1 time: 0.000000.
new max 30: 0 time 0.000000.
new max 27: 2 time 0.000000.
new max 24: 20 time 0.000000.
lower: 5, upper: 24, r: 2 time: 0.000000.
new max 21: 1, 20 time 0.000000.
lower: 7, upper: 21, r: 3 time: 0.000000.
lower: 9, upper: 21, r: 4 time: 0.030000.
lower: 11, upper: 21, r: 5 time: 0.480000.
lower: 13, upper: 21, r: 6 time: 7.250000.
lower: 15, upper: 21, r: 7 time: 89.440000.
lower: 17, upper: 21, r: 8 time: 920.830000.
lower: 19, upper: 21, r: 9 time: 8071.440000.
Final lower: 21, upper: 21, time: 61140.650000.
21
Time: 61140.650
```