ADDITIVE CODES OVER GF(4) AND THEIR APPLICATIONS

Zlatko Varbanov

Department of Mathematics and Informatics
University of Veliko Tarnovo, Bulgaria

Bayreuth, 21.04.2010

ADDITIVE CODES OVER $GF(q)$

Additive code C over $GF(q)$ of length n – additive subgroup of $GF(q)^n$ (if $x, y \in C \Rightarrow x + y \in C$)

Connections:

$\Rightarrow$ **Quantum codes (Calderbank, Rains, Shor, and Sloane)**

$\Rightarrow$ **combinatorial t-designs (Pless and Kim)**

$\Rightarrow$ **undirected graphs (Glynn; Schlingemann and Werner)**

$\Rightarrow$ **other combinatorial structures (Huffman, Gulliver, Parker)**

ADDITIVE CODES OVER $GF(4)$

$GF(4) = \{0, 1, \omega, \omega^2\}$, **and** $\omega^2 + \omega + 1 = 0$**.**

Additive code C *over* $GF(4)$ *of length* n – **additive subgroup of** $GF(4)^n$**. We call** C **an** $(n, 2^k)$ **code** $(0 \leq k \leq 2n)$**.**

Weight **of a codeword** $c \in C$ $(wt(c))$ **is the number of nonzero components of** c**.**

Minimum weight (distance):
$d = d(C) = min\{wt(c)|c \in C, c \neq 0\} \rightarrow (n, 2^k, d)$ **code.**

Generator matrix of C – $k \times n$ **matrix with entries in** $GF(4)$ **whose rows are a basis of** C**.**

Weight enumerator of C**:** $C(z) = \sum_{i=0}^{n} A_i z^i$

ADDITIVE CODES OVER $GF(4)$

Trace **map** $Tr : GF(4) \longrightarrow GF(2)$ **is given by** $Tr(x) = x + x^2$**.** **In particular** $Tr(0) = Tr(1) = 0$ **and** $Tr(\omega) = Tr(\omega^2) = 1$**.**

The *conjugate* **of** $x \in GF(4)$ **(denoted** $\bar{x}$**) is the following image of** x**:** $\bar{0} = 0, \bar{1} = 1$**, and** $\bar{\omega} = \omega^2$**.**

The *trace inner product* **of two vectors** $x = (x_1, x_2, \ldots, x_n)$**,** $y = (y_1, y_2, \ldots, y_n)$ **in** $GF(4)^n$ **is**

$$x \star y = \sum_{i=1}^{n} Tr(x_i \bar{y_i}) \tag{1}$$

ADDITIVE SELF-DUAL CODES

Dual **code** $(C^{\perp})$ $-$ $C^{\perp} = \{x \in GF(4)^n | x \star c = 0$ **for all** $c \in C\}$**.**

If C **is an** $(n, 2^k)$ **code, then** $C^{\perp}$ **is an** $(n, 2^{2n-k})$ **code.**

Self-orthogonal **additive code -** $C \subseteq C^{\perp}$

Self-dual **additive code -** $C = C^{\perp}$**; it is** $(n, 2^n)$ **code.**

Type II **code - additive self-dual code, all codewords have even weight**

Type I **code - additive self-dual code, some codewords have odd weight**

BOUNDS

Bounds on the minimum weight (Rains and Sloane)

$$d_I \leq \begin{cases} 2\lfloor n/6 \rfloor + 1, & n \equiv 0 \ (mod\ 6); \\ 2\lfloor n/6 \rfloor + 3, & n \equiv 5 \ (mod\ 6); \\ 2\lfloor n/6 \rfloor + 2, & \textbf{otherwise} \end{cases} \qquad (2)$$

$$d_{II} \leq \ 2\lfloor n/6 \rfloor + 2$$

A code that meets the appropriate bound is called *extremal*.

If the code is not extremal but no code of given type can exist with a larger minimum weight, the code is called *optimal*.

EQUIVALENCE

Equivalent **additive codes - C_1 and C_2 are equivalent if there is a map sending the codewords of C_1 onto the codewords of C_2 where the map consists of a permutation of coordinates, a scaling of coordinates by element of $GF(4)$, and conjugation of some of coordinates.**

Equivalence of two additive codes over $GF(4)$ – by operations on binary codes. The transformation from C into a binary code is done by applying the map

$\beta: 0 \to 000; 1 \to 011; \omega \to 101; \bar{\omega} \to 110 \mid (n, 2^k) \to [3n, k]_2$ **code**

$$G_4 = \begin{pmatrix} 1 & \omega \\ 0 & \bar{\omega} \end{pmatrix} \to G_2 = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

I.Bouyukliev - Q-Extension

EQUIVALENCE

An additive code that is not $GF(4)$-linear, can be equivalent to a $GF(4)$-linear code with respect to the definition of an equivalence of additive codes.

Example: two additive self-dual $(2, 2^2)$ codes with generator matrices

$$\begin{pmatrix} 1 & 1 \\ \omega & \omega \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ \omega & \bar{\omega} \end{pmatrix}$$

The first code is $GF(4)$-linear but the second is not. But they are equivalent by conjugation of the second column of the generator matrix of the first code.

PRELIMINARY RESULTS

$\Rightarrow$ **All extremal codes** $2 \leq n \leq 7$ **–** ***Höhn, 1996***

$\Rightarrow$ **All extremal codes** $n = 8, 9, 11, 12$ **–** ***Gaborit, Huffman, Kim, and Pless, 2001***

$\Rightarrow$ **All additive self-dual codes** $n \leq 12$ **–** ***Parker and Danielsen, 2005***

$\Rightarrow$ **All extremal codes** $n = 13, 14$**; some codes** $15 \leq n \leq 21$ **–** ***Varbanov, 2006***

$\Rightarrow$ **Some codes** $15 \leq n \leq 28$ **with an automorphism of odd prime order –** ***Huffman, 2007***

PROBLEM: To construct/classify extremal ASD codes over $GF(4)$ **of length** $n \geq 15$**.**

ASYMPTOTIC NOTATIONS

O–notation:

$O(g(n)) = \{f(n)$: **there exist positive constants** c **and** n_0 **such that** $0 \leq f(n) \leq c.g(n)$ **for all** $n = n_0\}$**.**

Ω–notation:

$\Omega(g(n)) = \{f(n)$: **there exist positive constants** c **and** n_0 **such that** $0 \leq c.g(n) \leq f(n)$ **for all** $n = n_0\}$**.**

Θ–notation:

$\Theta(g(n)) = \{f(n)$: **there exist positive constants** c_1, c_2**, and** n_0 **such that** $0 \leq c_1.g(n) \leq f(n) \leq c_2.g(n)$ **for all** $n = n_0\}$**.**

CONSTRUCTIVE ALGORITHMS (Shortening)

Gaborit, Huffman, Kim, and Pless – 2001

Let C be an additive self-dual $(n, 2^n, d)$ code → by this algorithm an additive self-dual code of length $n-1$ can be constructed.

Let G be a generator matrix of C. Choose any column of G, say the i^{th} one. The *shortened code of C on coordinate i*, **denoted C', is the code with generator matrix G' obtained from G by eliminating one row of G with a nonzero entry in column i and then eliminating column i.**

C' – additive self-dual $(n-1, 2^{n-1}, d' \geq d-1)$ code.

Example:

$$G = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \omega & \omega & \omega \end{pmatrix} \rightarrow G' = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \textbf{ or } \begin{pmatrix} 1 & 1 \\ \omega & \omega \end{pmatrix}$$

SHORTENING (Complexity)

Weight enumerator – an additive code over $GF(4)$ consists of all $GF(2)$-linear combinations of the rows of the generator matrix. Therefore, to calculate weight enumerator we can use binary Gray code.

To reduce a column (one or two nonzero entries) – $O(n^2)$

To reduce all columns – $O(n^3)$

To find the minimum distance of any code – 2^n operations (using the binary Gray code)

Complexity: $\underline{O(n^3.2^n)}$

ALGORITHMS (Lengthening)

C **is an additive self-dual** $(n-1, 2^{n-1}, d)$ **code** → **by this algorithm we can construct a self-dual** $(n, 2^n, d')$ **code.**

If x **is a vector with entries in** $GF(4)$

$$G' = \left(\begin{array}{c|c} & 0 \\ G & \textbf{or} \\ & \omega \\ \hline \boldsymbol{x} & 1 \end{array}\right)$$

generates an additive self-dual $(n, 2^n, d')$ **code** C' **with minimum distance** $d' \leq d+1$**.**

Example:

$$G = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, x = [\omega\ \omega] \quad \rightarrow \quad G' = \begin{pmatrix} 0 & 1 & \omega \\ 1 & 1 & 0 \\ \omega & \omega & 1 \end{pmatrix}$$

LENGTHENING (Complexity)

$x \in \mathbb{F}_4^n \Rightarrow 4^n$ **possibilities**

To calculate the column vector $- O(n^2)$

To construct all possible codes of length $n + 1 - O(n^2.4^n)$

To find the minimum distance of any code $- 2^n$ **operations**

Complexity: $\underline{O(n^2.4^n.2^n) = O(n^2.2^{3n})}$

CIRCULANT CODES

An additive circulant $(n, 2^n)$ code has a circulant generator matrix of the following form:

$$A = \begin{pmatrix} a_0 & a_1 & a_2 & \dots & a_{n-2} & a_{n-1} \\ a_{n-1} & a_0 & a_1 & a_2 & \dots & a_{n-2} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_2 & \dots & a_{n-2} & a_{n-1} & a_0 & a_1 \\ a_1 & a_2 & \dots & a_{n-2} & a_{n-1} & a_0 \end{pmatrix}$$

An additive cyclic code over $GF(4)$ is generated by one or two generators.

$$a = (1\ 0\ \bar{\omega}\ \omega\ 1) \Rightarrow A = \begin{pmatrix} 1 & 0 & \bar{\omega} & \omega & 1 \\ 1 & 1 & 0 & \bar{\omega} & \omega \\ \omega & 1 & 1 & 0 & \bar{\omega} \\ \bar{\omega} & \omega & 1 & 1 & 0 \\ 0 & \bar{\omega} & \omega & 1 & 1 \end{pmatrix}$$

CIRCULANT CODES (Complexity)

No known part of the generator matrix is necessary;

Non-exhaustive search;

At most 4^n possibilities for the vector $a \in GF(4)^n$ that generates the code;

To find the minimum distance of any code $-$ 2^n operations

Complexity of the algorithm: $O(4^n.2^n) = O(2^{3n})$.

GRAPH CODES

Graph code – **additive self-dual code over $GF(4)$ with generator matrix $\Gamma + \omega I$, where I is the identity matrix and Γ is the adjacency matrix of a simple undirected graph which must be symmetric with 0's along the diagonal.**

EXAMPLE:

$$\Gamma = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad G = \Gamma + \omega I = \begin{pmatrix} \omega & 1 & 1 \\ 1 & \omega & 0 \\ 1 & 0 & \omega \end{pmatrix}$$

Theorem (Schlingemann and Werner, 2002): *For any self-dual additive code, there is an equivalent graph code. This means that there is a one-to-one correspondence between the set of simple undirected graphs and the set of self-dual additive codes over $GF(4)$.*

MINIMUM DISTANCE OF A GRAPH CODE

Minimum distance – the special form of the generator matrix of a graph code makes it easier to find the distance of the code. If the generator matrix is given in this form it is not necessary to check all the codewords to find the minimum distance of the code.

If $s \in C$ and $wt(s) \leq e$ – then s is a linear combination of at most e rows of the generator matrix of a graph code C.

– I. Bouyukliev, V. Bakoev, „A method for efficiently computing the number of codewords of fixed weights in linear codes“, Discrete Applied Mathematics, Volume 156 , Issue 15 (2008),Pages: 2986-3004

TYPE II CODES AND ANTI-EULERIAN GRAPHS

The graphs corresponding to *Type II* codes have a special property.

Theorem **(Parker and Danielsen, 2006)** *Let Γ be the adjacency matrix of the graph $\mathcal{G}$. The code C generated by $G = \Gamma + \omega I$ is of Type II if and only if $\mathcal{G}$ is anti-Eulerian, i.e., if all its vertices have odd degree.*

Other property:

Extremal $Type\ II$ **codes of given length** n **have a unique weight enumerator (Gaborit and Pless, 2001)**

'2–EXTENDING' OF GRAPH CODES

Parker and Danielsen, 2006

Let G be a generator matrix of $(n, 2^n, d)$ code C. First, we add an arbitrary n-dimensional binary vector x as $(n+1)^{th}$ row and $(n+1)^{th}$ column. To each obtained matrix G' we add as $(n+2)^{th}$ row and $(n+2)^{th}$ column the following vector y:

$$y_i = (1 + \sum_{j=1}^{n+1} g_{i,j})\ mod\ 2,\ \ i \neq j,\ \ 1 \leq i \leq n+1$$

EXAMPLE:

$$G = \begin{pmatrix} \omega & 1 \\ 1 & \omega \end{pmatrix}, x = (0\ 1) \Rightarrow G' = \begin{pmatrix} \omega & 1 & 0 \\ 1 & \omega & 1 \\ 0 & 1 & \omega \end{pmatrix},\ G'' = \begin{pmatrix} \omega & 1 & 0 & 0 \\ 1 & \omega & 1 & 1 \\ 0 & 1 & \omega & 0 \\ 0 & 1 & 0 & \omega \end{pmatrix}$$

’2–EXTENDING’ OF GRAPH CODES (Complexity)

To construct all possible matrices $G' - O(2^n)$

To construct the corresponding matrices $G'' - O(n^2)$

To check the minimum distance of any code – $\sum_{i=1}^{d-1}\binom{n}{i}$ operations

Complexity:

$$O(n^2.2^n\sum_{i=1}^{d-1}\binom{n}{i})$$

Note: this algorithm can be used only for construction of *Type II* codes.

LENGTHENING OF GRAPH CODES

Lemma: (ZV, 2006) If G is a generator matrix of a graph code of length n and x is a binary vector

$$G' = \left(\begin{array}{c|c} G & x^t \\ \hline x & \omega \end{array} \right) \tag{3}$$

is a generator matrix of a graph code of length $n+1$.

Example:

$$G = \begin{pmatrix} \omega & 1 & 0 \\ 1 & \omega & 1 \\ 0 & 1 & \omega \end{pmatrix}, x = [1\ 0\ 1] \quad \rightarrow \quad G' = \begin{pmatrix} \omega & 1 & 0 & 1 \\ 1 & \omega & 1 & 0 \\ 0 & 1 & \omega & 1 \\ 1 & 0 & 1 & \omega \end{pmatrix}$$

LENGTHENING OF GRAPH CODES (Complexity)

$x \in \mathbb{F}_2^n \Rightarrow$ **all possible codes of length** $n+1 - O(2^n)$

To check the minimum distance (at least d**)** $- \sum_{i=1}^{d-1} \binom{n}{i}$ **operations**

Complexity:

$$O(2^n \sum_{i=1}^{d-1} \binom{n}{i})$$

Here $d \approx n/3 < n/2$ **and**

$$\sum_{i=1}^{d-1} \binom{n}{i} < \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \leq 2^{n-1}$$

The number of known extremal(optimal) ASD codes over $GF(4)$

n	d_I	**Old bound**	**New bound**	**n**	d_{II}	**Old bound**	**New bound**
13	**5**	≥ 9 **(1)**	**85845**	**13**	–	–	–
14	**6**	**? (2)**	**2**	**14**	**6**	1020 **(3)**	1020
15	**6**	≥ 4 **(1)**	$\geq$ **2118**	**15**	–	–	–
16	**6**	≥ 15 **(1)**	$\geq$ **8371**	**16**	**6**	≥ 28 **(1)**	$\geq$ **112**
17	**7**	≥ 1 **(2)**	$\geq$ **2**	**17**	–	–	–
18	**7**	? **(2)**	$\geq$ **2**	**18**	**8**	≥ 1 **(2)**	≥ 1
19	**7**	≥ 4 **(1)**	$\geq$ **17**	**19**	–	–	–
20	**8**	≥ 3 **(1)**	≥ 3	**20**	**8**	≥ 5 **(1)**	≥ 5
21	**8**	≥ 1 **(2)**	$\geq$ **2**	**21**	–	–	–

(1)– Gulliver and Kim, 2004; (2)– Huffman, 2005; (3)– Danielsen and Parker, 2005.

ADDITIVE CIRCULANT GRAPH CODES

Additive circulant graph (ACG) code **– a code corresponding to graph with circulant adjacency matrix.**

Example:

$$B = \begin{pmatrix} \omega & 1 & 0 & 0 & 1 \\ 1 & \omega & 1 & 0 & 0 \\ 0 & 1 & \omega & 1 & 0 \\ 0 & 0 & 1 & \omega & 1 \\ 1 & 0 & 0 & 1 & \omega \end{pmatrix}$$

The generating vector has the following property: $b_i = b_{n-i}, \forall\ i = 1, \ldots, n-1$, **and** $b_0 = \omega$.

Then, the entries in the generator matrix of ACG code depend only on the coordinates $(b_1, b_2, \ldots, b_{\lfloor n/2 \rfloor})$.

THE ALGORITHM

INPUT: positive integers n and d $(1 < d < n)$.

OUTPUT: all possible ACG codes of length n and minimum distance $\geq d$.

- **STEP 1: If n is even, take a binary vector $g^{(0)} = (g_1, g_2, \ldots g_{\frac{n}{2}})$ and extend it to a vector $g = (\omega, g_1, g_2, \ldots, g_{\frac{n}{2}-1}, g_{\frac{n}{2}}, g_{\frac{n}{2}-1}, \ldots, g_2, g_1)$. If n is odd then $g^{(0)} = (g_1, g_2, \ldots g_{\frac{n-1}{2}})$, and $g = (\omega, g_1, g_2, \ldots, g_{\frac{n-1}{2}}, g_{\frac{n-1}{2}}, \ldots, g_2, g_1)$**
- **STEP 2: Construct a circulant matrix G (a generator matrix of an ACG code) with generating vector g.**
- **STEP 3: Compute all linear combinations of $1, 2, ..., d-1$ rows of G and check their weights. If all weights are $\geq d$ then the minimum distance is at least d.**
- **STEP 4: If $g^{(0)}$ is not all-one vector $-$ $g^{(0)} = g^{(0)} + 1$, Step 1.**
- **END.**

Complexity:

$$O(2^{\lfloor n/2 \rfloor} \sum_{i=1}^{d-1} \binom{n}{i})$$

RESULTS

ACG codes of length $13 \leq n \leq 36$ for the maximum reached d

n	d	number	n	d	number	n	d	number
13	5	2	21	7	11	29	11	1
14	6	3	22	8	14	30	12	$\geq$ 1
15	6	2	23	8	2	31	10	62
16	6	6	24	8	51	32	10	108
17	7	1	25	8	31	33	10	76
18	6	52	26	8	210	34	10	$\geq$ 144
19	7	4	27	8	140	35	10	$\geq$ 1
20	8	2	28	10	1	36	10	$\geq$ 4

GRAPH CODES OVER $GF(q^2)$

A *graph code* C is an additive code over $GF(q^2)$ that has a generator matrix of the form $G = \Gamma + \omega I$, where I is the identity matrix, ω is a primitive element of $GF(q^2)$, and Γ is the adjacency matrix of a undirected q-weighted graph.

Example: **A graph code over $GF(9)$**

$$\Gamma = \begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 2 & 0 \\ 2 & 2 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \quad G = \Gamma + \omega I = \begin{pmatrix} \omega & 1 & 2 & 1 \\ 1 & \omega & 2 & 0 \\ 2 & 2 & \omega & 1 \\ 1 & 0 & 1 & \omega \end{pmatrix}$$

Theorem **(Danielsen, 2008): Every self-dual additive code over $GF(q^2)$ is equivalent to a graph code.**

Classification of all self-dual additive codes over $GF(9), GF(16)$, and $GF(25)$ up to lengths 8, 6, and 6, respectively (Danielsen, 2008).

A RELATION TO QUANTUM CODES

MAIN PROBLEM: To construct good quantum codes (this is difficult problem, in general).

Let V be complex Hilbert space (tensor product of N smaller spaces). Single elements of V will represent pure states of quantum computer. A quantum code Q will then be a subspase of V (G.Nebe, E.Rains, N.Sloane - „Self-Dual Codes and Invariant Theory“).

Theorem: **Let C be an additive self-orthogonal $(n, 2^{n-k})$ code over $GF(4)$ such that no codewords with weight $< d$ in $C^{\perp} \backslash C$. Then, there exists a quantum code with parameters $[[n, k, d]]$. (Calderbank, Rains, Shor, and Sloane, 1998)**

QUANTUM COMPUTING

- **Quantum computing is a relatively new interdisciplinary field that has recently attracted many researchers from physics, mathematics, and computer science.**
- **The main idea of quantum computing is to utilize the laws of quantum physics to perform fast computations.**
- **Quantum information is represented by the states of quantum mechanical systems.**
- **Since the information–carrying quantum systems will inevitably interact with their environment, one has to deal with decoherence effects that tend to destroy the stored information.**
- **Hence, it is infeasible to perform quantum computations without introducing techniques to remedy this problem**

QUANTUM INFORMATION

$V = \otimes^N(\mathbb{C}^2) = \mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \ldots \otimes \mathbb{C}^2$, $dimV = 2^N$.

The tensor factors $\mathbb{C}^2$ are often called quantum bits (qubits). A qubit has two possible states, labelled $|0\rangle$ and $|1\rangle$.

$$|0\rangle = \binom{1}{0},\ |1\rangle = \binom{0}{1},\ \alpha|0\rangle + \beta|1\rangle = \binom{\alpha}{\beta}$$

Unlike a classical bit, a qubit can be in a superposition of $|0\rangle$ and $|1\rangle$. The state of a general qubit can be denoted $\alpha|0\rangle + \beta|1\rangle$ $(\alpha, \beta \in \mathbb{C})$, with $|\alpha|^2 + |\beta|^2 = 1$.

Here $|\alpha|^2$ being the probability of getting the result $|0\rangle$ when measuring the qubit, and $|\beta|^2$ the probability of getting a $|1\rangle$.

Several qubits form *quantum register*. The state of a two-qubit register can be denoted $\alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$.

QUANTUM INFORMATION

A quantum state can not be copied, i.e., there is no operation that takes $|\phi\rangle$ to $|\phi\phi\rangle$, where $|\phi\rangle$ is any quantum state.

$|\phi\rangle$, $|\psi\rangle$ **– quantum states;**

Copying operation:
$|\phi\rangle \to |\phi\phi\rangle$; $\quad |\psi\rangle \to |\psi\psi\rangle$; $\quad |\phi\rangle + |\psi\rangle \to |\phi\phi\rangle + |\psi\psi\rangle$

But by tensor product:
$|\phi\rangle + |\psi\rangle \to (|\phi\rangle + |\psi\rangle) \otimes (|\phi\rangle + |\psi\rangle) = |\phi\phi\rangle + |\psi\psi\rangle + |\phi\psi\rangle + |\psi\phi\rangle$

QUANTUM CODES

The space of errors to a single qubit is spanned by the four unitary matrices (Pauli operators):

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \; X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \; Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \; Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Any error on a single qubit, $|\phi\rangle \rightarrow E|\phi\rangle$, may be expressed as a linear combination of the Pauli matrices.

$$|\phi\rangle \rightarrow (aI + bX + cZ + dY)|\phi\rangle = a|\phi\rangle + bX|\phi\rangle + cZ|\phi\rangle + dY|\phi\rangle$$

The minimum distance d of a quantum code, is the minimum weight error operator that gives an errored state not orthogonal to the original state, and therefore not guaranteed to be detectable (Ashikhmin, Knill, 2002).

QUANTUM CODES

Let G_n be a finite group generated by the matrices in the error basis and $E \in G_n$ be an element of that group ($E = E_1 \otimes E_2 \otimes \ldots \otimes E_n$, E_i - an error on a single qubit).

Weight of an error E: $wt(E) = |\{E_i \neq I\}|$

A quantum code Q is said to have minimum distance d if and only if it can detect all errors in G_n of weight less than d, but cannot detect some error of weight d. Also, such a code can correct $\lfloor (d-1)/2 \rfloor$ errors in G_n.

An additive quantum $[[N, k, d]]$ code is a 2^k dimensional subspace of V with minimum distance d.

ADDITIVE AND LINEAR CODES OVER $GF(4)$

Every $GF(4)$-linear code is an additive code but the opposite is not true.

Example:

$$[6,3,4]_4 : \begin{pmatrix} 1 & 0 & 0 & 1 & \omega & \omega \\ 0 & 1 & 0 & \omega & 1 & \omega \\ 0 & 0 & 1 & \omega & \omega & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 1 & \omega & \omega \\ \omega & 0 & 0 & \omega & \bar{\omega} & \bar{\omega} \\ 0 & 1 & 0 & \omega & 1 & \omega \\ 0 & \omega & 0 & \bar{\omega} & \omega & \bar{\omega} \\ 0 & 0 & 1 & \omega & \omega & 1 \\ 0 & 0 & \omega & \bar{\omega} & \bar{\omega} & \omega \end{pmatrix} : (6, 2^6, 4)$$

The additive code C is linear iff c is a codeword then ωc is also a codeword.

ADDITIVE AND LINEAR CODES OVER $GF(4)$

Hermitian inner product
$u.v = \sum_{i=1}^{n} u_i\overline{v_i}$ – $u, v \in GF(4)^n$, $\overline{v_i} = v_i^2$

A linear code over $GF(4)$ is self-orthogonal (with respect to the Hermitian inner product) if and only if it is additive self-orthogonal (with respect to the trace inner product) code – (Calderbank et al., 1998)

$\Rightarrow$ If C is a Hermitian self-orthogonal linear $[n, k, d]$ code over $GF(4)$ with dual distance $d^{\perp}$ then there exists a quantum error-correcting $[[n, n-2k, d^{\perp}]]$ code.

RESULTS FOR QUANTUM CODES

There exist quantum codes with parameters:

- $[[24, 8, 5]]$ **(s.o.** $[24, 8, 10]_4$ **code,** $d^{\perp} = 5$**)** – **old bound** $d = 4$**;**
- $[[28, 10, 6]]$ **(s.o.** $[28, 9, 12]_4$ **code,** $d^{\perp} = 6$**)** – **old bound** $d = 5$**;**
- $[[30, 12, 6]]$ **(s.o.** $[30, 9, 12]_4$ **code,** $d^{\perp} = 6$**)**– **old bound** $d = 5$**;**
- $[[30, 16, 5]]$ **(s.o.** $[30, 7, 16]_4$ **code,** $d^{\perp} = 5$**)** – **old bound** $d = 4$**;**
- $[[32, 6, 8]]$ **(s.o.** $[32, 13, 12]_4$ **code,** $d^{\perp} = 8$**)** – **old bound** $d = 7$**;**
- $[[34, 24, 4]]$ **(s.o.** $[34, 5, 22]$ **code,** $d^{\perp} = 4$**)** – **old bound** $d = 3$

GRAPH CODES AND GF(4)-LINEAR CODES

A graph code cannot be GF(4)-linear (van den Nest, 2005).

Could be a graph code equivalent to GF(4)-linear code?

The conjugation of some of the columns is equivalent to the following:

- **Let A be a binary diagonal matrix, and $G = \Gamma + \omega I$ is a generator matrix of an additive self-dual code C;**
- **Then $G' = \Gamma + A + \omega I$ generates a code C' that is equivalent to C.**

C' **is linear iff** $\Gamma^2 + A\Gamma + \Gamma A + \Gamma = I$ **(van den Nest, 2005)**

COMBINATORIAL DESIGNS

$t-(v,k,\lambda_t)$ **design (**t**-design) – a pair** $(V,\mathcal{B})$ **where** V **is a set of** v **points and** $\mathcal{B}$ **is a set of** b **blocks each containing** k **different points, each point being contained in** r **different blocks and every** t **different points being contained in exactly** λ_t **blocks.**

$\lambda_t \binom{v}{t} = b \binom{k}{t}$ **and for** $0 \le s \le t$**, each** $t-(v,k,\lambda_t)$ **design is** $s-(v,k,\lambda_s)$ **design.**

$$\lambda_s = \lambda_t \binom{v-s}{t-s} / \binom{k-s}{t-s}$$ **(**$\lambda_1 = r$ **and** $\lambda_0 = b$**).**

COMBINATORIAL DESIGNS

Two t-designs $(V_1, \mathcal{B}_1)$ and $(V_2, \mathcal{B}_2)$ are isomorphic if there exists a bijection $\alpha : V_1 \rightarrow V_2$ such that $\mathcal{B}_1\alpha \rightarrow \mathcal{B}_2$.

An automorphism is an isomorphism of a t-design with itself.

The set of all automorphisms of a t-design D forms a group, the (full) automorphism group – $Aut(D)$.

***Support* of a vector x is the set of all coordinate positions of x such that any nonzero position is denoted by 1.**

Example: $x = (\omega 0 1 \bar{\omega} 0 \omega 1) \rightarrow (1011011)$

ADDITIVE CODES OVER $GF(4)$ AND DESIGNS

THEOREM (Kim and Pless, 2003): Let $n_i = 6m + 2(i-1)$ with $m \geq 1$ any integer and $i = 1, 2,$ or 3. Let C be an extremal additive $Type\ II$ $(n_i, 2^{n_i})$ code over $GF(4)$ with minimum weight $d = 2m + 2 \geq 6$. Then the supports of the vectors of each weight w in C where $A_w \neq 0$ and $d \leq w \leq n_i$ hold a $(7-2i)$-design with possibly repeated blocks.

THEOREM (Kim and Pless, 2003):

The set of supports of the odd minimum weight vectors in a code shortened from a linear extremal even self-dual code hold a simple design (without repeated blocks).

RESULTS FOR DESIGNS

1020 nonequivalent $(14, 2^{14}, 6)$ ***Type II*** **codes, weight enumerator:** $1 + 273z^6 + 2457z^8 + 7098z^{10} + 6006z^{12} + 549z^{14}$**.**

Codewords of weight 6 – 3-design with possibly repeated blocks, $\lambda_3 \binom{14}{3} = 273 \binom{6}{3} \Rightarrow \lambda_3 = 15$**.**

1020 nonisomorphic $3-(14,6,15)$ **designs – 490 designs with repeated blocks and 530 without repeated blocks.**

Number of $3-(14,6,15)$ **designs** D **with** $|Aut(D)| = \alpha$

α	**1**	**2**	**3**	**4**	**6**	**8**	**12**	**18**
number	**625**	**258**	**27**	**38**	**27**	**13**	**7**	**1**

α	**21**	**24**	**28**	**36**	**48**	**84**	**168**	**2184**
number	**1**	**16**	**1**	**1**	**1**	**1**	**1**	**2**

RESULTS FOR DESIGNS

There are 5 known *Type II* additive $(20, 2^{20}, 8)$ **codes (Gulliver and Kim, 2004)**

– the set of supports of weight 8 vectors hold a 3-design with possibly repeated blocks. Weight enumerator: $1+1710z^8+20976z^{10}+\ldots+141360z^{18}+6444z^{20}$

$\Rightarrow \lambda = 1710\binom{20}{3}/\binom{8}{3} = 84$**.**

$\Rightarrow$ **5 nonisomorphic** $3-(20,8,84)$ **designs (three designs with repeated blocks and two simple designs) – group order 20, 40, 6840, 2880, and 3840.**

Two codes are equivalent to the known GF(4)-linear $[20,10,8]$ **self-dual codes. By shortening:**

$\Rightarrow$ **two nonequivalent simple** $2-(19,7,28)$ **designs** D_1 **and** D_2**,** $|Aut(D_1)| = 144,\ |Aut(D_2)| = 192$**.**

THANKS FOR YOUR ATTENTION!