Classification of optimal $(v, 4, 1)$ optical orthogonal codes with $v \leq 76$

Tsonka Baicheva   Svetlana Topalova

Institute of Mathematics and Informatics
Bulgarian Academy of Sciences

September, 2010
Novosibirsk, Russia

Introduction I

F.R.K. Chung, J.A. Salehi and V.K. Wei, Optical orthogonal codes: design, analysis and applications, *IEEE Trans. Inform. Theory* **35**, 595–604, 1989.

- Optical code-division multiple-access communication systems
- Mobile radio
- Frequence-hopping spread spectrum communications
- Constructing protocol-sequence sets for the M-active-out-of-T users collision channel without feedback
- Radar and sonar signal design
- Public key algorithm for optical communication based on lattice cryptography

Introduction II

Optical orthogonal codes with specific parameters are closely related to

- Constant-weight error-correcting codes
- Difference sets
- Cyclic partial designs
- Well-correlated binary sequences

Basic definitions I

- Z_v the ring of integers modulo v

Definition

A $(v, k, \lambda_a, \lambda_c)$ optical orthogonal code (OOC) can be defined as a collection $\mathcal{C} = \{C_1, ..., C_s\}$ of k-subsets (*codeword-sets*) of Z_v such that any two distinct translates of a codeword-set share at most λ_a elements while any two translates of two distinct codeword-sets share at most λ_c elements:

$$|Ci \cap (Ci + t)| \leq \lambda_a, \quad 1 \leq i \leq s, \quad 1 \leq t \leq v - 1 \tag{1}$$

$$|Ci \cap (Cj + t)| \leq \lambda_c, \quad 1 \leq i < j \leq s, \quad 0 \leq t \leq v - 1 \tag{2}$$

- (1) is called the auto-correlation property
- (2) is called the cross-correlation property

Basic definitions II

- The size of $\mathcal{C}$ is the number s of its codeword-sets.
- A (v, k, λ, λ) OOC is also denoted by (v, k, λ) OOC.

$C = \{c_1, c_2, \ldots, c_k\}$ is a codeword-set
$\triangle' C$ is the multiset of the values of the differences
$c_i - c_j,\ i \neq j,\ i, j = 1,\ 2,\ \ldots,\ k$
$\triangle C$ is the underlying set of $\triangle' C$

- Autocorrelation property $\Rightarrow$ at most λ_a differences are the same
- Cross-correlation property $\Rightarrow$ if $\lambda_c = 1$ then $\Delta C_1 \bigcap \Delta C_2 = \emptyset$ for two codeword-sets C_1 and C_2 of the $(v, k, \lambda_a, 1)$ OOC

Basic definitions III

Definition

Two $(v, k, \lambda_a, \lambda_c)$ optical orthogonal codes are equivalent if they can be mapped to one another by an automorphism of Z_v and (or) replacement of codeword-sets by some of their translates.

Bound for the size of $(v, k, 1)$ OOC

$$s \leq \left\lfloor \frac{(v-1)}{k(k-1)} \right\rfloor$$

- $(v, k, 1)$ OOCs for which $s = \left\lfloor \frac{(v-1)}{k(k-1)} \right\rfloor$ are called optimal
- If $s = \dfrac{(v-1)}{k(k-1)}$ the $(v, k, 1)$ OOC is called perfect

A perfect $(v, k, 1)$ OOC corresponds to

- a cyclic 2-(v,k,1) design
- a cyclic (v,k,1) difference family

Investigations about $(v, k, 1)$ OOC I

- K. Chen and L. Zhu, Existence of (q, k, 1) difference families with q a prime power and k = 4, 5. *Combin. Des.* **7**, 21–30, 1999.
- M. Buratti, Cyclic designs with block size 4 and related optimal optical orthogonal codes, *Des. Codes Cryptogr.* **26**, 111–125, 2002.
- Y. Chang, R. Fuji-Hara and Y. Miao, Combinatorial constructions of optimal optical orthogonal codes with weight 4, *IEEE Trans. Inform. Theory*, **49**, 1283–1292, 2003.
- R.Julian, R. Abel and M. Buratti, Some progress on (v, 4, 1) difference families and optical orthogonal codes, *J. Combin. Theory*, Ser. A 106, 59–75, 2004.
- X. Wang and Y. Chang, Further results on (v,4,1)-perfect difference families, *Discrete Math.*, **310**, Issues 13-14, 1995–2006, 2010.
- M. Buratti and A. Pasotti, Further progress on difference families with block size 4 or 5, *Des. Codes Cryptogr.* Published online, doi:10.1007/s10623-009-9335-6.

Investigations about $(v, k, 1)$ OOC II

- An optimal $(v, 4, 1)$ OOC exists for all $v \leq 1212, v \neq 25$
- Classification results for small v are only known for cyclic $2-(v, 4, 1)$ designs, namely for the perfect $(v, 4, 1)$ OOCs for $v = 37, 49$ and 61.

Classification result about $(v, 4, 2)$

W. Chu and C.J. Colbourn, Optimal (n, 4, 2)- OOC of small order, *Discrete Math.* **279**, 163–172, 2004.

- A table of optimal $(v, 4, 2)$ OOCs with $v \leq 44$ (with 3 possible exceptions) is presented.
- Construction by an algorithm based on the maximum clique search problem.

Our result

We classify up to equivalence optimal $(v, 4, 1)$ OOCs with $v \leq 76$

Our approach:
ordering all possibilities for codeword-sets with respect to the action of the automorphisms of the cyclic group of order v, and then applying the well-known techniques of back-track search with minimality test on the partial solutions

P.Kaski and P.Östergård, *Classification algorithms for codes and designs*, Springer, Berlin, 2006.

Classification algorithm I

- We relate to each codeword-set $C = \{c_1, c_2, c_3, c_4\}$ a codeword-set vector $\vec{C} = (c_1, c_2, c_3, c_4)$ such that $c_1 < c_2 < c_3 < c_4$
- If we replace a codeword-set $C \in \mathcal{C}$ with a translate $C + t \in \mathcal{C}$, we obtain an equivalent OOC

w.l.o.g. we assume that each codeword-set vector of the optimal $(v, 4, 1)$ OOC is lexicographically smaller than the codeword-set vectors of its translates

- This means that $c_1 = 0$

Classification algorithm II

We create an array L of all 4-dimensional vectors over Z_v which might become codeword-set vectors

- We construct the vectors of L in lexicographic order
- To each vector we apply the automorphisms $\varphi_i, i = 1, 2, ...m-1$ of Z_v and if some of them maps it to a smaller vector, we do not add this vector since it is already somewhere in the array
- If we add the current vector $\vec{C}$ to the list, we also add after it the $m-1$ vectors to which $\vec{C}$ is mapped by $\varphi_i, i = 1, 2, ...m-1$.

This way we obtain the array L whose elements $L_x, x = 0, 1, ..., f$ are all the possible codeword-set vectors

Classification algorithm III

Codewordsets with suitable autocorrelation

$\mathbf{L_0}$

$L_1 = \varphi_1 L_0$

$L_2 = \varphi_2 L_0$

$\vdots$

$L_{m-1} = \varphi_{m-1} L_0$

$\mathbf{L_m}$

$L_{m+1} = \varphi_1 L_m$

$L_{m+2} = \varphi_2 L_m$

$\vdots$

$L_{2m-1} = \varphi_{m-1} L_m$

$\vdots$

$\mathbf{L_{im}}$

$L_{im+1} = \varphi_1 L_m$

$L_{im+2} = \varphi_2 L_m$

$\vdots$

$L_{(i+1)m-1} = \varphi_{m-1} L_m$

Classification algorithm IV

It is possible for two different automorphisms to map a codeword-set vector to one and the same codeword-set vector.

Example

The automorphism group of Z_{30} is of order 8
$\varphi_0(a) = a$, $\varphi_1(a) = 7a$, $\varphi_2(a) = 11a$, $\varphi_3(a) = 13a$, $\varphi_4(a) = 17a$, $\varphi_5(a) = 19a$, $\varphi_6(a) = 23a$, $\varphi_7(a) = 29a$, where $a \in Z_{30}$.

$$\vec{C_1} = (0,1,3,22) \stackrel{\varphi_1}{\rightarrow} (0,7,21,4) \stackrel{+26}{\rightarrow} (26,3,17,0) \rightarrow (0,3,17,26) = \vec{C_2}$$

$$\vec{C_1} = (0,1,3,22) \stackrel{\varphi_3}{\rightarrow} (0,13,9,16) \stackrel{+17}{\rightarrow} (17,0,26,3) \rightarrow (0,3,17,26) = \vec{C_2}$$

- We keep for each possible codeword-set vector L_x the smallest number a, such that $L_x = L_y$ and $y = a\ (mod\ m)$
- We keep this a in place of the first codeword-set element c_1, which is always 0

This way for each x we can directly obtain the smallest y, such that L_y is obtained by applying on L_x a given automorphism of Z_v.

We construct the OOC choosing the codeword-sets among the elements of L by backtrack search until we find the s codeword-sets

$$L_{x_1}, L_{x_2}, ..., L_{x_s}$$

- We choose the $r+1$-st element $L_{x_{r+1}}$ $(x_{r+1} > x_r)$ of the codeword-set to have no common differences with the previous r ones
- When we add the $r+1$-st codeword-set number x_{r+1}, we also find the $r+1$ numbers obtained by applying $\varphi_i, i = 1, 2, ..., m$ to the current partial solution and sort them
- If the obtained array is lexicographically smaller than the current one, it means that an equivalent sub-code with $r+1$ codeword-sets has already been considered, and we look for the next possibility for the $r+1$-st codeword-set.

Classification results

Table: Inequivalent optimal (v,4,1) OOCs

v	s	OOCs	v	s	OOCs	v	s	OOCs
26	2	1	43	3	1772	60	4	7585950
27	2	4	44	3	3208	61p	5	18132
28	2	4	45	3	12428	62	5	20736
29	2	11	46	3	9999	63	5	529996
30	2	41	47	3	20692	64	5	409632
31	2	42	48	3	51510	65	5	3774498
32	2	64	49p	4	224	66	5	6512840
33	2	196	50	4	336	67	5	18814608
34	2	181	51	4	5530	68	5	27675160
35	2	378	52	4	6382	69	5	153524880
36	2	731	53	4	28672	70	5	204850952
37p	3	2	54	4	56064	71	5	425759570
38	3	12	55	4	213662	72	5	979134632
39	3	96	56	4	263102	73p	6	1426986
40	3	86	57	4	1105056	74	6	2140556
41	3	338	58	4	1011104	75	6	59992260
42	3	998	59	4	2575944	76	6	42145856

Classification results

Table: Inequivalent optimal (v,4,1) OOCs

v	s	OOCs	v	s	OOCs	v	s	OOCs
26	2	1	43	3	1772	60	4	7585950
27	2	4	44	3	3208	**61p**	**5**	**18132**
28	2	4	45	3	12428	62	5	20736
29	2	11	46	3	9999	63	5	529996
30	2	41	47	3	20692	64	5	409632
31	2	42	48	3	51510	65	5	3774498
32	2	64	**49p**	**4**	**224**	66	5	6512840
33	2	196	50	4	336	67	5	18814608
34	2	181	51	4	5530	68	5	27675160
35	2	378	52	4	6382	69	5	153524880
36	2	731	53	4	28672	70	5	204850952
37p	**3**	**2**	54	4	56064	71	5	425759570
38	3	12	55	4	213662	72	5	979134632
39	3	96	56	4	263102	73p	6	1426986
40	3	86	57	4	1105056	74	6	2140556
41	3	338	58	4	1011104	75	6	59992260
42	3	998	59	4	2575944	76	6	42145856