Classification of Resolvable 2-(14,7,12) and 3-(14,7,5) Designs

Petteri Kaski*, Luis B. Morales†, Patric R. J. Östergård‡,
David A. Rosenblueth† and Carlos Velarde†
*Department of Computer Science and Engineering,
Helsinki University of Technology,
P.O. Box 5400, 02015 HUT, Finland.
†IIMAS, Universidad Nacional Autónoma de México,
Apdo. Postal 70-221, México, DF, 04510, México.
‡Department of Electrical and Communications Engineering,
Helsinki University of Technology,
P.O. Box 3000, 02015 HUT, Finland.

Abstract

The resolvable 2-$(14, 7, 12)$ designs are classified in a computer search: there are $1, 363, 486$ such designs, $1, 360, 800$ of which have trivial full automorphism group. Since every resolvable 2-$(14, 7, 12)$ design is also a resolvable 3-$(14, 7, 5)$ design and vice versa, the latter designs are simultaneously classified. The computer search utilizes the fact that these designs are equivalent to certain binary equidistant codes, and the classification is carried out with an orderly algorithm that constructs the designs point by point. As a partial check, a subset of these designs is constructed with an alternative approach by forming the designs one parallel class at a time.

Keywords: *backtrack search; equidistant code; orderly algorithm; resolvable t-design*

1 Introduction

We use the following standard notations. A t-(v, k, λ) *design* is a pair $(X, \mathcal{B})$, where X is a v-set of *points*, and $\mathcal{B}$ is a collection of k-subsets of X, called *blocks*, such that each t-subset of X occurs in exactly λ blocks. A design is said to be *resolvable* if the blocks can be partitioned into a collection of *parallel classes*, each of which partitions the point set. A partition of the blocks into parallel classes is a *resolution* of the design.

Two t-(v,k,λ) designs are *isomorphic* if there exists a bijection between the point sets that maps blocks onto blocks; such a bijection is an *isomorphism*. An *automorphism* of a design is an isomorphism of the design onto itself. The *(full) automorphism group* of a design consists of all of its automorphisms with composition of permutations as the group operation. Two resolutions of a design are isomorphic if there exists an isomorphism of the underlying designs that maps one resolution onto the other.

In this paper we classify, using computer search, all resolvable 2-(14, 7, 12) designs up to isomorphism. The following theorem of Alltop [1] shows that our classification is also a complete classification of the resolvable 3-$(14,7,5)$ designs.

Theorem 1 *A resolvable t-$(2k,k,\lambda)$ design with t even is a resolvable $(t+1)$-$(2k,k,\lambda')$ design with $\lambda' = \lambda(k-t)/(2k-t)$ and vice versa.*

Prior to this classification only one resolvable 2-$(14,7,12)$ design was known [3, 5].

Further definitions and background results on resolvable designs are given in Section 2. Of central importance is a correspondence between resolutions of designs and certain error-correcting codes due to Semakov and Zinov'ev [10]. Section 3 outlines the classification algorithm, which is based on the algorithm developed to establish nonexistence of a resolvable 2-$(15,5,4)$ design in [4]. The classification, which proceeds point by point, is discussed in Section 4. There are exactly 1,363,486 nonisomorphic resolvable 2-$(14,7,12)$ designs (and resolvable 3-$(14,7,5)$ designs by Theorem 1). Finally, to gain confidence in correctness of the classification, we perform an independent partial verification in Section 5, by forming the designs one parallel class at a time.

2 Preliminaries

Given a t-(v,k,λ) design, we denote the number of blocks in which a point occurs by r and the number of blocks by b. Then, a straightforward double counting argument gives

$$vr = bk, \qquad r\binom{k-1}{t-1} = \lambda\binom{v-1}{t-1}. \tag{1}$$

Furthermore, it is easy to see that a necessary condition for resolvability is that k divides v. In addition, a resolution consists of r parallel classes, each of which consists of v/k blocks. A 2-$(14,7,12)$ design has $b=52$ and $r=26$.

A resolution of a resolvable t-$(2k, k, \lambda)$ design is unique since every block can form a parallel class only with its complement block. (In general a resolution need not be unique. For example, there are 4 nonisomorphic resolvable 2-$(15, 3, 1)$ designs; these have 7 nonisomorphic resolutions [5].)

A *(binary block) code* of length n is a nonempty set $C \subseteq Z_2^n$, where $Z_2 = \{0, 1\}$. The elements $x = (x_1, \ldots, x_n) \in C$ are called *codewords*. The *cardinality* of a code is the number of codewords it contains. The *Hamming distance* between two codewords $x, y \in C$ is the quantity

$$d_H(x, y) = |\{j \in \{1, \ldots, n\} \ : \ x_j \neq y_j\}|.$$

The *minimum distance* of a code C is the minimum Hamming distance between pairs of codewords, taken over distinct codewords of C. A code is *equidistant* if all pairs of distinct codewords have the same Hamming distance. An (n, M, d) code has length n, cardinality M, and minimum distance d. The next result, which is called the *Plotkin bound* (originally proved in [9]; a generalization to the q-ary case can be found in [2]), is of central importance in our classification.

Theorem 2 *If there exists an (n, M, d) code and $2d > n$, then*

$$M \leq \frac{2d}{2d - n}. \tag{2}$$

If equality holds, an (n, M, d) code is equidistant, and the coordinate values are evenly distributed in each coordinate (so M must then be even).

Two (n, M, d) codes are *equivalent* if their codewords are related by a permutation of the coordinates and a permutation of the coordinate values $\{0, 1\}$. More formally, two (n, M, d) codes, the codewords of which we assume to be labelled as $\{y^{(1)}, \ldots, y^{(M)}\}$ and $\{z^{(1)}, \ldots, z^{(M)}\}$, are equivalent if there exist permutations $\sigma \in \mathrm{Sym}(\{1, \ldots, M\})$, $\pi \in \mathrm{Sym}(\{1, \ldots, n\})$, and $\mu_1, \ldots, \mu_n \in \mathrm{Sym}(\{0, 1\})$ such that

$$z_j^{(i)} = \mu_j(y_{\pi(j)}^{(\sigma(i))}) \tag{3}$$

holds for all $i = 1, \ldots, M$ and $j = 1, \ldots, n$.

The following construction is a special case of a more general result of Semakov and Zinov'ev [10]. Label the points $X = \{x_1, \ldots, x_{2k}\}$ and the blocks $\mathcal{B} = \{B_1, \ldots, B_b\}$ of a resolvable 2-$(2k, k, \lambda)$ design so that blocks B_{2j-1}, B_{2j} form a parallel class of the resolution for all $j = 1, \ldots, r$. Then, we obtain a $(r, 2k, r - \lambda)$ equidistant code by defining the codewords of the code by the rule

$$y_j^{(i)} = \begin{cases} 0 & \text{if } x_i \in B_{2j-1} \\ 1 & \text{if } x_i \in B_{2j} \end{cases} \tag{4}$$

for all $i = 1, \ldots, 2k$ and $j = 1, \ldots, r$. Clearly, the labelling of the blocks determines the code equivalence class representative obtained.

Conversely, if $t = 2$, we obtain from (1) that equality holds in (2) for an $(r, 2k, r - \lambda)$ code. Thus, the above construction can be reversed, that is, from an arbitrary $(r, 2k, r - \lambda)$ equidistant code we can construct a resolvable 2-$(2k, k, \lambda)$ design by labelling the codewords and applying (4) in the reverse direction; the labelling of the codewords then determines the design isomorphism class representative obtained.

The previous construction demonstrates that the isomorphism classes of resolvable 2-$(2k, k, \lambda)$ designs are in a one-to-one correspondence with the equivalence classes of $(r, 2k, r - \lambda)$ equidistant codes. We apply this correspondence in the classification algorithm outlined in the next section.

3 The search

We use backtrack search to construct exactly one representative from each isomorphism class of resolvable 2-$(14, 7, 12)$ designs, that is, equivalence class of $(26, 14, 14)$ codes. The search is based on the algorithm developed in [4]; see that paper for further details.

A partial solution in the search is an equidistant code of length 26 and minimum distance 14. Rejection of equivalent codes is achieved by using the canonicity predicate from [4]. Briefly, a code satisfies the canonicity predicate if and only if it is the lexicographic minimum in its equivalence class determined by (3).

The search has two stages. The first stage proceeds by adding one codeword at a time to the partial solution. It is required that the added codeword is lexicographically greater than any of the codewords in the current partial solution. The addition of a codeword is followed by an application of the canonicity predicate to the augmented partial solution; if the augmented solution fails to satisfy the predicate, then it is not considered further.

The second stage determines all completions of a partial solution $C \subseteq Z_2^{26}$ to an equidistant $(26, 14, 14)$ code by performing an exhaustive search for all $(14 - |C|)$-cliques in the compatibility graph of C. The vertices of the *compatibility graph* of C consist of all codewords $x \in Z_2^{26}$ that are (a) lexicographically greater than any codeword in C; and (b) for which $d_H(x, y) = 14$ holds for all $y \in C$. Two vertices x, x' are connected by an edge if and only if $d_H(x, x') = 14$. After the compatibility graph has been constructed, the maximum clique algorithm described in [8] is employed to locate the $(14 - |C|)$-cliques. Each $(26, 14, 14)$ code obtained from C and the vertices of a $(14 - |C|)$-clique is filtered using the canonicity predicate to produce a collection of representatives for the equivalence classes.

The codewords required in both stages are constructed coordinatewise by backtrack search. In constructing the codewords we take advantage of Theorem 2, which shows that each coordinate of an $(26, 14, 14)$ code has exactly 7 zeros and 7 ones. So, the first 7 codewords added to the code must contain a zero in the lexicographically most significant coordinate of the code, and the subsequent 7 codewords must contain a one.

In generating the $(26, 14, 14)$ codes, the threshold between the first and the second stages of the algorithm was set to 8 codewords, up to which the first stage was applied. The second stage was then used to augment a code with 6 further codewords.

The classification was completed in about two CPU days on a workstation with 1GHz AMD Athlon CPU. The maximum order of a compatibility graph was 1,280.

4 Results

Table 1 contains a classification of the resolvable 2-$(14, 7, 12)$ designs by full automorphism group order. Space restrictions prevent us from explicitly listing but a few of the designs here; anyone with further interest in the designs should contact the first author so that a listing of the designs of interest can be prepared. In particular, a listing of the resolvable 2-$(14, 7, 12)$ designs with nontrivial full automorphism group is readily available electronically at

⟨URL:`http://www.tcs.hut.fi/~pkaski/res-14-7-12.html`⟩.

Theorem 3 *There are* $1,363,486$ *nonisomorphic resolvable* 2-$(14, 7, 12)$ *designs;* $1,360,800$ *of these have trivial full automorphism group.*

Table 2 lists four resolvable 2-$(14, 7, 12)$ designs that admit compact description using their automorphism groups. For each design the table lists generator permutations for the full automorphism group and representatives of the parallel class orbits that under the action of the full automorphism group form the unique resolution of the design. (Only one block from each parallel class is listed since the other block in the parallel class is obtained as the complement of the listed block with respect to the point set $\{1, \ldots, 14\}$.) Each of the designs admits the 13-cycle $(2 \cdots 14)$ as an automorphism, so the designs are easily reconstructed even by hand calculation.

Table 1: The resolvable 2-$(14, 7, 12)$ designs.

$\lvert\mathrm{Aut}(\mathcal{D})\rvert$	Nrd
1	1,360,800
2	1,819
3	748
4	63
6	37
8	1
12	13
13	1
24	1
39	2
156	1
Total	1,363,486

Table 2: Four resolvable 2-$(14, 7, 12)$ designs with large automorphism group order.

$\lvert\mathrm{Aut}(\mathcal{D})\rvert$	Orbit representatives	Generators for $\mathrm{Aut}(\mathcal{D})$
156	{1,2,3,4,5,8,12}	(2 3 4 5 6 7 8 9 10 11 12 13 14) (3 13 6 7 5 9 14 4 11 10 12 8)
39	{1,2,3,4,9,12,13} {1,2,3,5,7,9,10}	(2 3 4 5 6 7 8 9 10 11 12 13 14) (3 5 11)(4 8 7)(6 14 12)(9 10 13)
39	{1,2,3,4,6,7,10} {1,2,3,4,6,9,11}	(2 3 4 5 6 7 8 9 10 11 12 13 14) (3 11 5)(4 7 8)(6 12 14)(9 13 10)
13	{1,2,3,4,5,9,13} {1,2,3,5,8,9,13}	(2 3 4 5 6 7 8 9 10 11 12 13 14)

5 A partial verification of the classification

We shall now look at a partial verification of the earlier result. Instead of proceeding point by point, we now proceed parallel class by parallel class. The approach used is closely related to that in [7], from where we have borrowed some of the terminology.

We define the *parallel class intersection matrix* (PCIM) of two distinct parallel classes $p = \{B_{p1}, B_{p2}\}$ and $q = \{B_{q1}, B_{q2}\}$ on 14 points as the 2×2 matrix $A(p, q) = (a_{ij}(p, q))$, where $a_{ij}(p, q) = |B_{pi} \cap B_{qj}|$. Since a resolvable 2-$(14, 7, 12)$ design has 26 parallel classes, for each parallel class

$\{B_{p1}, B_{p2}\}$, there are 25 PCIMs with respect to the other parallel classes. Obviously, for any two parallel classes p and q, $a_{i1}(p,q) + a_{i2}(p,q) = 7$ for $i = 1, 2$. The blocks in these parallel classes can therefore be ordered so that the matrix $A(p,q)$ is one of

$$\begin{bmatrix} 7 & 0 \\ 0 & 7 \end{bmatrix}, \quad \begin{bmatrix} 6 & 1 \\ 1 & 6 \end{bmatrix}, \quad \begin{bmatrix} 5 & 2 \\ 2 & 5 \end{bmatrix}, \quad \begin{bmatrix} 4 & 3 \\ 3 & 4 \end{bmatrix},$$

and we say that the parallel classes meet in Type I, II, III, and IV, respectively.

Let $\{B_{p1}, B_{p2}\}$ be any parallel class of a resolvable 2-$(14,7,12)$ design. Let x_1, x_2, x_3, and x_4 be the number of matrices from the 25 PCIMs of Type I, II, III, and IV, respectively, with respect to the other parallel classes. It follows by counting pairs of points occurring in a block of the fixed parallel class that

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &= 25, \\ 42x_1 + 30x_2 + 22x_3 + 18x_4 &= 231. \end{aligned}$$

These equations have two solutions: $x_1 = 0$, $x_2 = 1$, $x_3 = 0$, $x_4 = 24$; and $x_1 = x_2 = 0$, $x_3 = 3$, $x_4 = 22$. In particular, note that $x_1 = 0$ for both solutions, so no parallel classes can meet in Type I. We summarize the result in the following theorem.

Theorem 4 *A necessary existence condition for a resolvable* 2-$(14,7,12)$ *design is that, for each parallel class p, the* 25 *PCIMs with respect to the other parallel classes agree with one of the patterns:*

(i) one matrix is of Type II and 24 *matrices are of Type IV,*

(ii) three matrices are of Type III and 22 *matrices are of Type IV.*

Based on Theorem 4 we divide resolvable 2-$(14,7,12)$ designs into two classes: designs of Type 1 have at least one parallel class that agrees with (i), and designs of Type 2 have parallel classes that agree with (ii) only.

We shall now construct all resolvable 2-$(14,7,12)$ designs of Type 1, which by definition contain two blocks whose intersection has cardinality six. This is carried out in a parallel class by parallel class backtrack search.

By combinatorial arguments one may restrict the search to start from the structures in Figure 1. These structures show, w.l.o.g., the distribution of the points among all parallel classes in the six points where two blocks meet.

Figure 1: Initial structures for the search.

012345		012345		012345		012345	
012345		012345		012345		012345	
0123	45	0123	45	0123	45	0123	45
0124	35	0123	45	0123	45	0124	35
0135	24	0145	23	0145	23	0125	34
0245	13	0245	13	0145	23	0345	12
1345	02	1345	02	2345	01	1345	02
2345	01	2345	01	2345	01	2345	01
012	345	012	345	012	345	013	245
013	245	013	245	013	245	013	245
014	235	014	235	014	235	014	235
014	235	014	235	015	234	014	235
015	234	015	234	023	145	015	234
015	234	015	234	024	135	015	234
023	145	023	145	024	135	023	145
023	145	024	135	024	135	023	145
024	135	024	135	025	134	024	135
025	134	025	134	025	134	024	135
025	134	025	134	025	134	025	134
034	125	034	125	034	125	025	134
034	125	034	125	034	125	034	125
034	125	034	125	034	125	034	125
035	124	035	124	035	124	035	124
035	124	035	124	035	124	035	124
045	123	035	124	035	124	045	123
045	123	045	123	045	123	045	123

In the backtrack search, symmetries are taken into account to speed up the search. Isomorph rejection of final solutions is carried out by looking for blocks that intersect in six points and record the intersection pattern with the rest of the blocks (cf. Figure 1), combined with an isomorphism check using the graph automorphism program *nauty* [6]. A total of 541,192 nonisomorphic resolvable 2-$(14, 7, 12)$ designs of Type 1 is found in this search; the order of the automorphism groups of these are presented in Table 3. The CPU time employed in the search was about 500 hours on a 800 MHz PC computer.

The numbers in Table 3 agree with the numbers obtained by counting the Type 1 designs in the complete classification, which gives an independent partial verification of the complete classification.

Table 3: The resolvable 2-$(14, 7, 12)$ designs of Type 1.

$\lvert\mathrm{Aut}(\mathcal{D})\rvert$	Nrd
1	461,119
2	646
3	187
4	33
6	34
8	1
12	11
13	1
24	1
156	1
Total	462,034

Acknowledgements

Research of the first and third author was supported in part by the Academy of Finland under grants 44517 and 100500.

References

[1] W. O. Alltop, Extending t-designs, *J. Combin. Theory Ser. A* **18** (1975), 177–186.

[2] G. T. Bogdanova, A. E. Brouwer, S. N. Kapralov, and P. R. J. Östergård, Error-correcting codes over an alphabet of four elements, *Des. Codes Cryptogr.* **23** (2001), 333–342.

[3] M. Hall, Jr., *Combinatorial Theory*, Blaisdell, Waltham, 1967.

[4] P. Kaski and P. R. J. Östergård, There exists no (15,5,4) RBIBD, *J. Combin. Des.* **9** (2001), 357–362.

[5] R. Mathon and A. Rosa, 2-(v, k, λ) designs of small order, *The CRC Handbook of Combinatorial Designs* (C. J. Colbourn and J. H. Dinitz, eds.), CRC Press, Boca Raton, 1996, pp. 3–41.

[6] B. D. McKay, *nauty* user's guide (version 1.5), Technical report TR-CS-90-02, Computer Science Department, Australian National University, 1990.

[7] L. B. Morales and C. Velarde, A complete classification of (12,4,3)-RBIBDs, *J. Combin. Des.* **9** (2001), 385–400.

[8] P. R. J. Östergård, A fast algorithm for the maximum clique problem, *Discrete Appl. Math.* **120** (2002), 195–205.

[9] M. Plotkin, Binary codes with specified minimum distance, *IRE Trans. Inform. Theory* **6** (1960), 445–450.

[10] N. V. Semakov and V. A. Zinov'ev, Equidistant q-ary codes with maximal distance and resolvable balanced incomplete block designs, *Problemy Peredachi Informatsii* **4**, no. 2 (1968), 3–10. Translated from Russian in *Problems Inform. Transmission* **4**, no. 2 (1968), 1–7.