There is no 2-$(22, 8, 4)$ Block Design

Richard Bilous, Clement W. H. Lam, Larry H. Thiel,
Department of Computer Science and Software Engineering
Concordia University
Montreal, Québec, Canada H3G 1M8

(Ben) P. C. Li, G. H. John van Rees,
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba, Canada R3T 2N2

Stanisław P. Radziszowski
Department of Computer Science
Rochester Institute of Technology
Rochester, NY, 14623

Wolfgang H. Holzmann, Hadi Kharaghani
Department of Mathematics and Computer Science
University of Lethbridge
Lethbridge, Alberta, Canada T1K 3M4

August 4, 2006

Abstract

In this paper we show that a 2-$(22, 8, 4)$ design does not exist. This result was obtained by a computer search.

1 Introduction

In this paper, we assume that the reader is familiar with the basic definitions of a 2-(v, k, λ) design; see, for example, [6, p. 3–13]. For an introduction to the computational methods, see [6, p. 718–740]. For a summary of the history of this design, see [16].

A 2-(v, k, λ) design, is a set $\mathcal{X} = \{x_i\}_{i=1}^{v}$ of *points* together with a family $\mathcal{B} = \{B_j\}_{j=1}^{b}$ of k-subsets (called *blocks*) such that each pair of distinct points occurs in exactly λ blocks. A 2-(v, k, λ) design is also called a *balanced incomplete block design* or *BIBD.*

The number of blocks b in a BIBD is given by

$$b = \frac{v(v-1)}{k(k-1)}\lambda,$$

and every point is in exactly

$$r = \frac{v-1}{k-1}\lambda$$

blocks.

A BIBD is completely determined by its *incidence matrix* $A = (a_{i,j})_{v \times b}$ where

$$a_{i,j} = \begin{cases} 1 & if \quad x_i \in B_j, \\ 0 & if \quad x_i \notin B_j. \end{cases}$$

In 1938, Fisher and Yates [7] produced tables of parameters for small BIBDs. Their smallest undecided case in the number of varieties (v) was for the parameters 2-$(22, 8, 4)$.

In [8], Hamada and Kobayashi analysed the possible block intersection patterns in a 2-$(22, 8, 4)$ design. They found 9 types, and eliminated 5 of them. After some extensive computing, McKay and Radziszowski [14, 15] eliminated 2 of the remaining 4 types.

Another approach is to consider the possible automorphism groups of such a design. In [10, 12], Kapralov, Landgev and Tonchev showed that the full automorphism group of a 2-$(22, 8, 4)$ design is either a 2-group or the trivial group.

Yet another approach is to consider the vector space generated by the rows of the incidence matrix A of a 2-$(22, 8, 4)$ design over GF(2). This is the (n, k) *point code* C of the design, where the *length* n is the number of points, 33, and the *dimension* k is the dimension of the subspace generated by the rows of A over GF(2). The vectors in the code are called *codewords*. The *weight* of a codeword is the number of its non-zero components. A code is called *doubly-even* if it contains only codewords with weights divisible by 4. Two codewords are *orthogonal* to each other if the dot product of the two codewords is 0. A code is *self-orthogonal* if any codeword is orthogonal to any other codewords in the code.

In [9], Hall *et al* showed that the point code of a 2-$(22, 8, 4)$ design is a $(33, k)$ doubly-even self-orthogonal code with dimension k between 8 and 16. In [1, 5], Bilous and van Rees showed that it suffices to consider only the case $k = 16$ because any $(33, k)$ doubly-even self-orthogonal code with $k < 16$ is contained in a $(33, 16)$ doubly-even self-orthogonal code. Moreover, we only have to consider codes that do not contain a coordinate of zeros.

In [1, 3, 4, 5], Bilous and van Rees showed that there exists 594 inequivalent doubly-even self-orthogonal $(33, 16)$ codes with no coordinates of zeros. Without using computers, they proved that 116 of these codes cannot contain the incidence matrix of a 2-$(22, 8, 4)$ design. It was the intersection of several weight 4 codewords in these codes that made their elimination possible.

2 Search Results

In [2], Bilous gave a detailed description of how to search for such a design, given a code. In particular, computer searches up to that point eliminated 299 of the 478 remaining codes.

The fact that these codes contained a least several weight 4 codewords eliminated many of the potential weight 12 codewords from being in the incidence matrix of the 2-(22,8,4). So the searches went relatively quickly. There were 147 remaining codes that were also eliminated relatively quickly because again they contained more than 1 weight 4 codeword. Generally speaking, the more weight 4 codewords there were in a code, the quicker the code could be eliminated.

In Table 4 of [2], Bilous also gave estimates on the size of the search space for the most difficult 32 of the remaining 179 codes. These 32 codes are the ones with at most one weight 4 word. Together, they account for most of the search space.

However, some of the pruning ideas were dropped in the final version of the computer program. They were dropped because the computing time required to implement these pruning tests was found to be more than the amount of computing time saved. In particular, the following prunning ideas were dropped:

1. pruning using patterns under a weight 5 word (Section 6.3 of [2]), and

2. equivalent case processed pruning (Section 6.4 of [2]).

Column 4 of Table 1 gives the revised estimates of the size of the search space for the first 32 codes. These 32 codes account for 91% of the search space.

The actual computing was carried out in several universities and using a variety of computers. Most of the computation were carried out between September 2002 and October 2005. No massively parallel supercomputers were used. Instead, typically, a few hundred computers on local area departmental networks at different universities were run simultaneously. A typical computer was a 2 gigahertz Intel CPU running the linux operating system. To allow for simultaneous running of the program, each code was divided into many subcases. The number of subcases was chosen so that each subcase took roughly 1 day of CPU time. To avoid losing a complete run if the computer was accidentally rebooted, the computer program regularly wrote status information to a disk file. If the run had to be restarted, the program would read the latest status information and continue from that point. Moreover, the software package autoson [13], a tool developed by Brendan McKay, was used to coordinate the scheduling of running jobs over hundreds of computers.

Table 1 gives a summary of the actual runs for the first 32 codes. For a desciption of the meaning of a *left pattern*, please see [2]. Basically, it is a 22×4 or a 22×5 submatrix, depending on whether the code has or does not have a weight 4 word. The *worst pattern* is the identification of the left pattern which gives rise to the largest node count for the code in question. One can see that the actual node counts are very close to the estimates.

Please see <http://www.cs.umanitoba.ca/~umbilou1/2-22_8_4_design/> for a summary for all 478 codes. This summary can also be obtained by contacting either the first or second author.

The total computing amounts to over 263 CPU years. This ranks as one of the largest computational efforts on a single task ever completed. As a comparison, the computer search for a projective plane of order 10 [11] took only about 125 CPU days on a CRAY-1, and the performance of a CRAY-1 is roughly equivalent to a 1 gigahertz Intel CPU.

Unfortunately, after all this computing, no 2-(22, 8, 4) designs were found.

Theorem 1 *There is no 2-*$(22, 8, 4)$ *design.*

We wish to emphasize that this is a computer-based proof, and that given all the possibilities of software and hardware errors, it is highly desirable to have an independent verification of the results.

3 Future Research

The coding techniques that were developed can be applied to any 2-(v, k, λ) design in which r and λ have the same parity. In order to have the same situation as in this paper, add $r(\text{mod}4)$ columns of all 1's to the incidence matrix of the design to generate a doubly-even, self-orthogonal binary code. This code can then be investigated.

An interesting family of block designs to look at are the BIBD's with parameters 2-$(6\lambda - 2, 2\lambda, \lambda)$. For $\lambda = 3$, the enumeration of non-isomorphic designs are known (see [6]) and the enumeration of inequivalent doubly-even, self-orthogonal (25,12) binary codes could easily be done,. But it would be quite interesting to know which designs embed into which codes. Another interesting block design occurs when $\lambda = 8$ in the above family. It is not known whether this BIBD exists or not [6]. Of course, one can not hope to do a complete enumeration of this case but perhaps one could prove theoretically that no design embeds into the appropriate code or find a code with a design embedded into it.

Other block designs are also interesting with smaller parameters. For example, the 2-(40,10,3) and the 2-$(46, 15, 3)$ designs are still much bigger than the the 2-$(22, 8, 4)$ but may have nicer theoretical properties. One hopes for new necessary conditions for the existence of block designs. It is not clear which BIBD parameter set should be investigated next with these techniques.

The smallest, in the number of varieties, undecided BIBD parameter set becomes the 2-(40,10,3). It has 52 blocks and at present, it seems hopeless to try any backtracking type of computing on it.

Acknowledgements

G. H. J. van Rees acknowledges the support of NSERC Discovery Grant 003358-04.
C. W. H. Lam acknowledges the support of NSERC Discovery Grant 9373-02.
W. H. Holzmann and H. Kharaghani acknowledge the support of NSERC Discovery and RTI grants.

The authors wish to acknowledge Patteri Kaski for his careful reading of the various versions of this manuscript and his many suggestions for improvments.

The authors also wish to acknowledge the contribution made by Malcolm Greig via both his preprint, An improvement to Connor's Criterion, and many useful conversations with many of the authors.

References

[1] R. T. Bilous, *The Point Code of a* $(22, 33, 12, 8, 4)$- *Balanced Incomplete Block Design*, PhD thesis, University of Manitoba, 2001.

[2] R. T. Bilous, "Searching a $(33, 16)$ doubly-even code for a $(22, 33, 12, 8, 4)$-BIBD", *J. Combin. Math. Combin. Comput.*, 46(2003), 53–64.

[3] R. T. Bilous, G. H. J. van Rees, "An Enumeration of Binary Self-Dual Codes of Length 32", *Des. Codes Cryptogr.*, 26(2002), 61-86.

[4] R. T. Bilous, "An enumeration of binary self-dual codes of length 34,", *J. Combin. Math. Combin. Comput.*, to appear.

[5] R. T. Bilous, G. H. J. van Rees, "Self-Dual Codes and the (22,8,4) Balanced Incomplete Block Design", *J. of Combin. Des.*, 13(2005), 363–376.

[6] C. Colbourn and J. Dinitz, eds., CRC Handbook of Combinatorial Designs, CRC Press, Boca Raton, 1996.

[7] R. A. Fisher and F. Yates. *Statistical Tables for Biological, Agricultural and Medical Research*. Longman, London, 1^{st} edition, 1938.

[8] N. Hamada and Y. Kobayshi, "On the block structure of bib designs with parameters $v = 22$, $b = 33$, $r = 12$, $k = 8$, and $\lambda = 4$", *J. Combin. Theory, Ser. A* 24(1978), 75–83.

[9] M. Hall Jr., R. Roth, G. H. J. van Rees, and S. A. Vanstone, "On designs $(22, 33, 12, 8, 4)$", *J. Combin. Theory, Ser. A* 47(1988), 157–175.

[10] S. N. Kapralov, "Combinatorial 2-$(22, 8, 4)$ designs with automorphisms of order 3 that fix a point(Bulgarian)", In *Mathematics and mathematical Education (Bulgarian), Proc. of the XVI Spring Conference of Union of Bulgarian Mathematicians*, Sunny Beach, 453–458, Publ. House Bulgar. Acad. Sci., Sofia 1987.

[11] C. W. H. Lam, L. Thiel, and S. Swiercz, "The Non-existence of Finite Projective Planes of Order 10", *Canad J. Math.*, 41(1989), 1117–1123.

[12] I. Landgev and V. Tonchev, "Automorphisms of 2-$(22, 8, 4)$ designs", *Discrete Math.* 77(1989), 177–189.

[13] B. D. McKay, *autoson — a distributed batch system for UNIX workstation networks (version 1.3)*, Technical Report TR-CS-96-03, Joint Computer Science Technical Report Series, Australian National University, 1996; `http://cs.anu.edu.au/~bdm/autoson`.

[14] B. D. McKay and S. P. Radziszowski, "Towards deciding the existence of 2-$(22, 8, 4)$ designs", *J. Combin. Math. Combin. Comput.* 22(1996), 211–222.

[15] B. D. McKay and S. P. Radziszowski, "2-$(22, 8, 4)$ designs have no blocks of type 3", *J. Combin. Math. Combin. Comput.* 30(1999), 251–253.

[16] G. H. J. van Rees, "$(22, 33, 12, 8, 4)$-BIBD, an update", In *Computational and Constructive Design Theory*, pages 337–357. W.D. Wallis, Kluwer Academic Publ., 1996.

code	num wt 4	num left patterns	estimated node count	actual node count	worst pattern	run time hour
0	0	83	$4.03e+15$	$4.01e+15$	6.0	259588
1	0	83	$3.71e+15$	$3.76e+15$	4.1	161088
2	0	82	$7.52e+15$	$7.56e+15$	6.0	652247
3	0	50	$2.64e+15$	$2.68e+15$	6.0	373486
4	0	35	$2.55e+14$	$2.61e+14$	6.0	54942
5	0	51	$1.05e+15$	$1.01e+15$	5.2	293564
6	0	51	$3.13e+14$	$3.26e+14$	5.2	20696
7	0	31	$6.71e+14$	$6.72e+14$	6.0	139443
8	0	8	$2.65e+14$	$2.55e+14$	3.0	15758
9	0	31	$1.70e+14$	$1.74e+14$	6.0	10865
10	0	7	$2.85e+13$	$1.53e+13$	3.0	1758
11	1	4	$8.39e+14$	$7.77e+14$	2.0	30447
12	1	7	$2.32e+14$	$2.60e+14$	2.0	57075
13	1	7	$3.37e+14$	$2.33e+14$	2.1	30301
14	1	7	$1.26e+14$	$1.77e+14$	2.0	6820
15	1	7	$1.93e+14$	$1.74e+14$	2.0	22287
16	1	7	$1.67e+14$	$1.75e+14$	2.1	6973
17	1	7	$1.46e+14$	$1.96e+14$	2.0	7564
18	1	7	$1.08e+14$	$1.14e+14$	2.1	30695
19	1	3	$1.28e+14$	$1.94e+14$	2.0	25138
20	1	3	$1.26e+14$	$1.30e+14$	2.0	4937
21	1	3	$7.53e+13$	$9.78e+13$	2.0	3695
22	1	12	$4.29e+13$	$3.66e+13$	2.0	1552
23	1	12	$5.02e+13$	$3.09e+13$	2.0	1563
24	1	7	$5.60e+13$	$3.51e+13$	2.0	1936
25	1	7	$2.51e+13$	$2.16e+13$	2.0	1210
26	1	3	$3.37e+13$	$1.85e+13$	2.0	1039
27	1	7	$3.98e+12$	$3.44e+12$	2.0	192
28	1	7	$1.86e+12$	$1.10e+12$	2.0	60
29	1	4	$1.50e+12$	$8.19e+11$	2.0	45
30	1	3	$8.37e+11$	$5.75e+11$	2.0	31
31	1	3	$1.94e+11$	$1.75e+11$	2.0	9

Table 1: Summary of search statistics for the first 32 codes