Defining Set Spectra for Designs can have Arbitrarily Large Gaps

George Havas, Julie L. Lawrence and Colin Ramsay
School of Information Technology and Electrical Engineering
The University of Queensland, Brisbane, QLD 4072, Australia

Anne Penfold Street
Department of Mathematics
The University of Queensland, Brisbane, QLD 4072, Australia

Emine Şule Yazıcı
Department of Mathematics
Koç Üniversitesi, Rumeli Feneri yolu, 34450 Sariyer, İstanbul, Turkey

Abstract

A set of blocks which is a subset of a *unique* t-(v, k, λ) design D is a *defining* set of D. A defining set is *minimal* if it does not properly contain a defining set. Define the *spectrum* of minimal defining sets of D by $spec(D) = \{|M| : M$ is a minimal defining set of $D\}$. Call h a *hole* in $spec(D)$ if $h \notin spec(D)$, but there are minimal defining sets of D with cardinalities both larger and smaller than h. If $spec(D)$ does not contain a hole, then it is said to be *continuous*. Previously, the spectra of only a limited number of designs were known and all of these were continuous. The question "whether the spectrum is continuous for all designs" was raised by B. Gray et al. (*Discrete Mathematics* 261 (2003), 277–284). We describe a new algorithm which finds all minimal defining sets of t-(v, k, λ) designs. Using this algorithm we investigated the spectra for a variety of small designs, and found several examples of non-continuous spectra. We also derive some theoretical results which enable us to construct an infinite family of designs with arbitrarily large sequences of consecutive holes in their spectra.

KEYWORDS: block design, defining set, spectrum, combinatorial algorithm
MSC: 05B05, 05-04, 68W05

1 Introduction

A *block design* is a set of b k-subsets (called *blocks*) chosen from a set of v elements, V, such that every element of V occurs in exactly r blocks. If every t-subset of V belongs to exactly λ blocks, the design is said to be *t-balanced*, and is called a *t-design*, with parameters t-(v, k, λ).

A collection of blocks which is a subset of a *unique* t-(v, k, λ) design D is said to be a *defining set* of D (see [7, 8, 9]). A *minimal* defining set is a defining set, no proper subset of which is a defining set. A *smallest* defining set is a defining set such that no other defining set has fewer blocks. The *spectrum* of minimal defining sets of D is $spec(D) = \{|M| : M$ is a minimal defining set of $D\}$. We call h a *hole* in $spec(D)$ if $m < h < n$, $h \notin spec(D)$, and $m, n \in spec(D)$. If $spec(D)$ does not contain a hole then it is said to be *continuous*. A *gap* in $spec(D)$ is a maximal length sequence of consecutive holes.

In this paper we investigate the spectra of a variety of t-(v, k, λ) designs. Many of the designs we consider contain repeated blocks and so are multisets rather than sets. We will use the language of sets when talking about designs and defining sets, but these should be taken to be sets or multisets as appropriate. When we wish to emphasise that something is a multiset, we use brackets (i.e. $[\cdot]$) instead of braces (i.e. $\{\cdot\}$). For simplicity, we frequently omit braces and commas; in particular, blocks will be written as, for example, 0123 instead of $\{0, 1, 2, 3\}$.

Example 1.1 *Consider the* 2-$(9, 3, 2)$ *design* $D = [123, 123, 145, 145, 167, 167, 189, 189, 246, 248, 257, 259, 268, 279, 347, 349, 356, 358, 369, 378, 468, 479, 569, 578]$. *The following are some minimal defining sets of D:*

1. $M_1 = [123, 123, 145, 145, 167, 167, 246, 248, 268, 468]$
2. $M_2 = [145, 145, 189, 189, 248, 257, 268, 279, 347, 349, 468]$
3. $M_3 = [123, 123, 145, 145, 167, 167, 248, 257, 259, 268, 347, 468]$

D has no minimal defining sets of size less then 10 *or more than* 12*, so* $spec(D) = \{10, 11, 12\}$ *(note that D is #36 in Table 3).*

The question of whether there is a design with a hole in its spectrum or whether all spectra are continuous was raised in [5], after the work of [4, 5] showed that the 2-$(15, 3, 1)$ design associated with $PG(3, 2)$ has minimal defining sets of each size from 16 to 22. In this paper we introduce a new algorithm that finds all the minimal defining sets of a design. Using this algorithm we have found examples of designs that are not continuous and which exhibit a variety of types of holes. We also develop some theoretical results which enable us to construct an infinite family of designs with arbitrarily large gaps.

2 Minimal defining set algorithm

Greenhill [10] and Delaney [2] were the first to describe and implement algorithms to find smallest defining sets. These algorithms start by finding a lower bound, n, for the size of a smallest defining set of the design D, and then determining whether any set of n blocks from D is a defining set. If not, sets of $n+1$ blocks are considered, and this process is continued until a defining set is found. Various techniques, including searches for *trades* (i.e. differences between pairs of designs), are used to improve the efficiency of the process. Khodkar [13] used a different approach via linear programming techniques on lists of minimal trades to find and test candidate smallest defining sets.

These methods all require a *completion* programme; that is, a routine to enumerate all designs with the given parameters which contain a given set of blocks. A completion programme is a backtracking search, and various implementations are described in [2, 14, 21]. The algorithm we introduce does not require a completion routine.

Given a t-(v,k,λ) design D and a list $\mathcal{L}$ of all designs with the same parameters as D, our algorithm finds all minimal defining sets of D in one pass. $\mathcal{L}$ can be generated by applying all $v!$ permutations to a complete set of non-isomorphic t-(v,k,λ) designs (duplicate entries in $\mathcal{L}$ merely generate redundant work). For small cases it is convenient to generate $\mathcal{L}$ by using a completion routine with the empty set as its input, but this is not essential.

The central feature of our algorithm is an array of binary flags, one for each possible subset of D. These flags are initially set to "0", and the entries for subsets are tagged (i.e. set to "1") as they are proved not to be minimal defining sets. Below we give pseudocode for our algorithm, followed by some explanatory comments. For clarity, we give the case where D is a set, not a multiset. The output is a representative of each isomorphism class of minimal defining sets. We use S^c to denote the canonic version of S produced by McKay's isomorphism testing programme *nauty* [16].

Number the subsets of D and initialise the flag array F
FOR ALL D^* IN $\mathcal{L} - D$ DO
    $I \leftarrow D \cap D^*$; $F[I] \leftarrow 1$
FOR ALL $S \subseteq D$ DO
    IF $F[S] = 1$ THEN tag all subsets of S
FOR ALL $S \subseteq D$ DO
    IF $F[S] = 0$ THEN tag all proper supersets of S
$\mathcal{C} \leftarrow \emptyset$
FOR ALL $S \subseteq D$ DO
    IF $F[S] = 0$ AND $S^c \notin \mathcal{C}$ THEN { $\mathcal{C} \leftarrow \mathcal{C} \cup S^c$; PRINT S }

Subsets of D are conveniently numbered by treating the bit-vector representation as an integer, and using this to index F. The intersections I in line 3 are contained in more than one design, so these cannot be defining sets. A subset of a non-defining set cannot be a defining set, and these are tagged in line 5. At this point, since $\mathcal{L}$ is a complete list, every untagged subset is a defining set. (We have, in effect, enumerated all the trades in D; cf. [22].) A proper superset of a defining set cannot be a minimal defining set, and these are tagged in line 7. The untagged subsets are now all the minimal defining sets. We record the canonic defining sets in $\mathcal{C}$, and print a representative of each isomorphism class.

If D contains repeated blocks (i.e. is a multiset) the basic algorithm is unchanged. However, in our implementation, repeated blocks are represented multiple times in the bit-vector representation of sets. Thus a given multiset may have more than one representation as a set. If two subsets are equal as multisets than the corresponding subsets are called *equivalent*.

We wish to retain only one subset from each group of equivalent subsets. To achieve this, we augment the algorithm as follows. In line 3, each time we tag a subset we also find and tag its equivalents. Between lines 7 and 8, we loop over all $S \subseteq D$ and if S is untagged we tag its equivalents; thus, S is the representative of its group of equivalents.

3 Spectrum hole problem

In this section we describe some designs that do not have continuous spectra, resolving the question in [5] in the negative. In Subsection 3.1 we give examples of spectra produced by our algorithm that have single or multiple holes. In Subsection 3.2 we describe some theoretical results which enable us to construct an infinite family of designs with arbitrarily large gaps.

3.1 Some example spectra

When we applied our algorithm to some small designs we found several which do not have continuous spectra; we give some examples below. Examples 3.1, 3.2 and 3.5 are designs with a single hole, while Example 3.3 is a design with 2 holes. Example 3.4 has a continuous spectrum but is used in Section 3.2 to construct designs with arbitrarily large gaps. Note that the minimal defining sets given are representatives of isomorphism classes of defining sets. These results are all new apart from Example 3.4, where Gray [8] found the smallest defining set while Greenhill and Street [11] found the other minimal defining set and showed that there are no more.

Example 3.1 *Let D be the full 2-$(6,3,4)$ design. (There are four 2-$(6,3,4)$ designs.) That is, $D = \{012,\ 013,\ \ldots,\ 345\}$ is the set of all $\binom{6}{3}$ 3-subsets from the set $\{0,\ldots,5\}$. Then spec$(D) = \{6,7,8,10\}$. Up to isomorphism, there are 1, 2, 3 and 2 minimal defining sets of size 6, 7, 8 and 10 respectively. Representative minimal defining sets are:*

1. $\{012, 013, 014, 023, 024, 034\}$
2. $\{012, 013, 014, 015, 023, 024, 025\}$
3. $\{012, 013, 014, 023, 024, 123, 124\}$
4. $\{012, 013, 014, 015, 023, 035, 123, 124\}$
5. $\{012, 013, 014, 015, 035, 045, 123, 124\}$
6. $\{012, 013, 023, 025, 035, 123, 124, 134\}$
7. $\{024, 025, 034, 035, 045, 123, 124, 125, 134, 135\}$
8. $\{015, 024, 034, 035, 045, 123, 124, 125, 135, 234\}$

Example 3.2 *Let D be the 2-$(8,4,3)$ design #4 from Appendix A. Then spec$(D) = \{6,8\}$, and representatives are $\{0257,\ 0347,\ 0356,\ 1247,\ 1256,\ 1346\}$ and $\{0123,\ 0145,\ 0246,\ 0257,\ 0347,\ 1247,\ 1256,\ 1346\}$.*

Example 3.3 *Let D be the 2-$(8,4,6)$ design formed by taking two copies of design #1 in Appendix A. Then spec$(D) = \{12,14,16\}$. There are 7, 3 and 2 non-isomorphic defining sets of size 12, 14 and 16 respectively.*

Example 3.4 *Let $D = \{012,\ 034,\ 056,\ 078,\ 136,\ 147,\ 158,\ 238,\ 245,\ 267,\ 357,\ 468\}$ be the 2-$(9,3,1)$ design derived from the affine geometry $AG(3,2)$. Then spec$(D) = \{4,5\}$, and representative minimal defining sets are $M_1 = \{056,\ 078,\ 136,\ 147\}$ and $M_2 = \{012,\ 034,\ 056,\ 136,\ 147\}$.*

Example 3.5 *Let D be the 2-$(9,3,2)$ design formed by taking two copies of the 2-$(9,3,1)$ design in Example 3.4 (note that D is #1 in Table 3). Then spec$(D) = \{8,10\}$, and representatives are $[056,\ 056,\ 078,\ 078,\ 136,\ 136,\ 147,\ 147]$ and $[012,\ 012,\ 034,\ 034,\ 056,\ 056,\ 136,\ 136,\ 147,\ 147]$.*

3.2 Arbitrarily large gaps

Two or more designs can be combined to form a larger design. Here we exploit this fact to construct designs with arbitrarily large gaps in their spectra. We will need to manipulate multisets in our proofs, so we make the following definitions.

Definition 3.6 *A multiset of elements from a set A is a function $M : A \to \mathbb{N}$, where $\mathbb{N}$ is the natural numbers. (The intuition is that, for each element $x \in A$, $M(x)$ denotes the number of occurrences of x in M.)*

Definition 3.7 *Let M and N denote multisets over a set A. Then:*

1. *$M = N$ if and only if $M(x) = N(x)$, for all $x \in A$*
2. *$M \subseteq N$ if and only if $M(x) \leq N(x)$, for all $x \in A$*
3. *$(M + N)(x) = M(x) + N(x)$, for all $x \in A$*
4. *$(M - N)(x) = \max(0, M(x) - N(x))$, for all $x \in A$*

Let D be a t-(v, k, λ) design. Suppose that $D = D_1 + D_2$ where D_1 and D_2 are, respectively, t-(v, k, λ_1) and t-(v, k, λ_2) designs, with $\lambda = \lambda_1 + \lambda_2$. Then D is said to be *reducible*. The following result, relating the defining sets of D_1, D_2 and D, is similar to that in [7, Theorem 1.9].

Theorem 3.8 *Let S be a defining set of the reducible design $D = D_1 + D_2$. Then there exist multisets $S_1 \subseteq D_1$ and $S_2 \subseteq D_2$, defining sets of D_1 and D_2 respectively, such that $(S_1 + S_2) \subseteq S$.*

Proof: Let D and S be as defined. Partition S into multisets A and B so that $A + B = S$ where $A \subseteq D_1$ and $B \subseteq D_2$. Assume A does not contain a defining set of D_1. By definition, there is at least one design $D_1^* \neq D_1$ that contains A. So $D_1^* + D_2$ also contains S, and S is not a defining set. Thus A contains a defining set, say S_1, of D_1. Similarly, B contains a defining set, S_2, of D_2. So $S_1 + S_2 \subseteq A + B = S$. □

In Theorem 3.8, D_1 and D_2 need not be identical. Consideration of the case where $D_1 = D_2$ leads to our next result. Note that, if M is a multiset and α is a positive integer then we use αM as shorthand for $\sum_{i=1}^{\alpha} M$.

Theorem 3.9 *Let D be a t-(v, k, λ) design and consider the t-$(v, k, \alpha\lambda)$ design αD. Then any defining set S of αD contains α copies of some defining set R of D; that is, $\alpha R \subseteq S$.*

Proof: Let D and αD be as defined, and let S be a defining set of αD. Let $X = S - (\alpha - 1)D$, and observe that $X \subseteq D$ and $\alpha X \subseteq S$. If X does not contain a defining set of D then, by the same argument as in the proof of Theorem 3.8, S is not a defining set. So X contains a defining set of D, and S contains α copies of that defining set, since $\alpha X \subseteq S$. □

Corollary 3.10 *Let D be the 2-$(9,3,\lambda)$ design formed by taking λ copies of the 2-$(9,3,1)$ design in Example 3.4. Then any minimal defining set of D contains either λ copies of M_1 or λ copies of M_2.*

Theorem 3.11 *If D is the 2-$(9,3,\lambda)$ design formed by λ copies of the 2-$(9,3,1)$ in Example 3.4 then S is a minimal defining set of D if and only if $S = \lambda M_1$ or $S = \lambda M_2$, where M_1 and M_2 are as in Example 3.4. So spec$(D) = \{4\lambda, 5\lambda\}$.*

PROOF: By Corollary 3.10 the minimal defining sets of D contain either λM_1 or λM_2. We show that λM_1 and λM_2 are defining sets of D, and are thus the only minimal defining sets of D.

Let $S = \lambda M_1$, and consider the pair 01. This can only occur in a block as 012, so λ copies of this block are forced in the design. This now forces λ copies of the blocks 034 and 158. Now consider the pair 38. This can only occur in a block as 238, so λ copies of 238 are forced. This now forces λ copies of the blocks 357 and 468, and then λ copies of the blocks 245 and 267.

Let $S = \lambda M_2$. Then λ copies of the blocks 078 and 158 are immediately forced. Now consider the pair 45. This can only occur in a block as 245, so λ copies of 245 are forced. This now forces, in turn, λ copies of 357, λ copies of 238 and 267, and λ copies of 468. □

In contrast with Theorem 3.11, the containment in Theorem 3.9 may be proper; see Example 4.1.

4 Results

In this section we give the results obtained using our algorithm on some families of designs. In all cases only the sizes of the smallest defining sets, and example smallest defining sets, were previously known. We provide tables, after the appendices, giving the number of isomorphism classes of minimal defining sets of each size for each design. In these tables, the first column gives the design's label, the second column gives its automorphism group order, and the remaining columns give the number of isomorphism classes.

There are 36 non-isomorphic 2-$(9,3,2)$ designs (see [15, 18]). In [13], Khodkar finds the sizes of the smallest defining sets and gives one smallest defining set for each design. The design numbering in Table 3 matches that used in [13].

There are four non-isomorphic 2-$(8,4,3)$ designs [20]. The smallest defining sets were found in [7]. Our results are summarised in Table 1, and Appendix A gives the designs used and a complete list of their minimal defining sets.

Although we have not investigated all the 2-$(8,4,6)$ designs, we have computed *spec*(D) for the four such designs which are doublings of the 2-$(8,4,3)$ designs; call these $1'$, $2'$, $3'$ and $4'$.

Example 4.1 *The minimal defining sets of designs* $1'$, $2'$ *and* $4'$ *are all doublings of those for designs* 1, 2 *and* 4, *respectively. However design* $3'$ *has spec*$(3') = \{12, 13, 14, 18\}$, *with counts* $25, 2, 2$ *and* 4, *respectively. So two of the size six minimal defining sets for design* 3 *do not double up to give defining sets for design* $3'$. *Thus a multiple of a defining set of a design is not necessarily a defining set of the multiple design; i.e., the containment in Theorem 3.9 may be proper.*

There are eleven non-isomorphic 2-$(9,4,3)$ designs [1, 3, 23]. The smallest defining sets were found by Moran [17]. Our results are summarised in Table 2, where the design's labels match those in [17]. Note that Moran gives counts of the number of classes of smallest defining sets, and our counts match these.

There are three non-isomorphic 2-$(10,4,2)$ designs [19]. The smallest defining sets were found in [11]. Our results are summarised in Table 4, and Appendix B gives the designs used and a complete list of their minimal defining sets.

Up to isomorphism, there is a unique 2-$(11,5,2)$ design. The smallest defining sets are given in [8, 11]. It is known that there are no larger minimal defining sets, but this does not seem to be recorded in the literature. We have confirmed this using our algorithm. Take the design to be $D = \{01237$, 01456, 02589, $0368a$, $0479a$, $1248a$, $1359a$, 16789, 23469, $2567a$, $34578\}$. Then *spec*$(D) = \{5\}$, and representative minimal defining sets are $\{01237$, 01456, 02589, $0368a$, $1248a\}$ and $\{01456$, 02589, $0368a$, $1248a$, $1359a\}$.

5 Concluding remarks

We have described a new algorithm which allows us to find minimal defining sets. Our algorithm is simple, is straightforward to programme, and yields complete information on all minimal defining sets. We have tested our implementation on a selection of the designs already considered in the literature, and our results agree with the published results. Previous authors have concentrated on smallest defining sets, and very few non-trivial spectra were known; most of the spectra we give here are new. We hope to publish a census of the small designs in due course. Meanwhile, we have made our software, together with brief instructions and some examples, available on the web [12].

Our algorithm is perfectly general and will, in principle, work for any t-(v, k, λ) design. However, its time and space requirements are exponential. The running time depends on the size of the list of all designs (which scales, in general, as $v!$) and the size of the flag array (i.e. 2^b), while the space required is that for the flag array. The key feature of our algorithm is the array of flags representing the subsets of the design, and the elimination of subsets which are not defining sets. In our implementation we used intersections to prune the array, but any combination of techniques which identified all the non-defining sets would suffice. Note that the algorithm could be adapted to investigate defining sets in other types of combinatorial designs.

The theoretical results presented in Section 3.2 allow us to construct designs with arbitrarily large gaps in their spectra. However, they do not explain all the spectra we present, so further research is needed on defining sets of reducible designs. In particular, when does combining two or more designs (not necessarily identical) yield a reducible design with holes in its spectrum?

All the spectra with holes which we give are in reducible designs, apart from the 2-$(8, 4, 3)$ design D of Example 3.2. However, D is self-complementary and can be constructed from the 2-$(7, 3, 1)$ design [7]. The blocks which contain 0, with 0 deleted, form a 2-$(7, 3, 1)$ design, while the blocks not containing 0 form a 2-$(7, 4, 2)$ design (in fact, they are the complement of the 2-$(7, 3, 1)$). Since a design and its complement have the same (i.e. complementary) defining sets [7], D can be considered to be reducible to two copies of a smaller design. This raises the question: is there a non-reducible design with holes in its spectrum?

Our infinite family of designs with gaps in their spectra is parametrised by λ, with all the designs being multiples of a common design. Designs with non-continuous spectra seem to be common, and it would be nice to prove the existence of an infinite family parameterised by v or that they exist for all large enough v.

Acknowledgements: The authors are grateful to Brendan McKay for his help in incorporating *nauty* into their programmes. The authors were partially supported by the Australian Research Council, the ARC Centre for Complex Systems, and the Centre for Discrete Mathematics and Computing. The fifth author's involvement in this work commenced while she was a postdoctoral fellow at The University of Queensland in 2004.

A Defining sets of the 2-$(8, 4, 3)$ designs

Tables 5 and 6 give the blocks of designs #1 and #3, and #2 and #4, respectively, written vertically under the design's label. Representatives of the minimal defining set classes are written vertically, with "X" indicating that the block is in the defining set and "-" indicating that it is not. The headings on the defining set columns give the sizes of the defining sets.

B Defining sets of the 2-$(10, 4, 2)$ designs

Tables 7 and 8 give the blocks and defining sets for designs #1 and #3, and #2, respectively. The layout of the tables is the same as in Appendix A.

Table 1: Minimal defining set classes of the 2-$(8, 4, 3)$ designs

		Counts		
Design	|Aut|	6	7	8
1	48	7	3	2
2	12	32	2	-
3	21	27	-	-
4	1344	1	-	1

Table 2: Minimal defining set classes of the 2-$(9, 4, 3)$ designs

		Counts			
Design	|Aut|	6	7	8	9
M_1	144	-	-	25	9
M_2	16	-	-	209	81
M_3	2	-	-	1644	496
M_4	8	-	-	417	124
M_5	1	-	-	3222	812
M_6	2	-	-	1617	406
M_7	6	-	-	539	136
M_8	8	23	222	406	-
M_9	32	6	53	104	-
M_{10}	1	-	-	3204	724
M_{11}	9	-	-	356	82

Table 3: Minimal defining set classes of the 2-$(9, 3, 2)$ designs

Design	|Aut|	Counts						
		8	9	10	11	12	13	14
1	432	1	-	1	-	-	-	-
2	24	7	2	85	14	2	-	-
3	32	3	2	61	50	12	1	-
4	8	-	12	134	154	47	15	-
5	24	1	10	94	77	2	2	-
6	6	9	15	357	430	169	12	-
7	4	11	51	471	1031	523	40	2
8	2	-	57	629	2016	877	70	4
9	2	-	57	662	1745	915	79	3
10	4	-	28	335	890	449	53	3
11	2	-	6	359	1806	1387	132	4
12	8	-	3	162	384	165	17	2
13	4	-	-	56	612	967	45	-
14	108	-	2	29	27	11	3	-
15	4	-	47	406	1829	1295	144	6
16	2	4	104	932	3293	2621	225	16
17	1	-	52	972	5705	5409	533	16
18	2	-	11	466	3038	3099	336	14
19	4	-	18	331	1385	1133	119	13
20	3	-	10	209	1657	1610	158	19
21	1	-	12	786	5234	5933	523	22
22	6	-	-	23	494	1375	121	1
23	18	-	8	73	505	484	75	3
24	6	-	22	234	1587	1754	238	4
25	8	2	16	216	1040	1343	152	5
26	1	-	48	1170	8057	10829	1641	49
27	1	-	84	1231	7687	9520	1599	58
28	1	-	22	768	7059	11085	1728	46
29	6	5	20	273	1334	1377	248	6
30	6	-	-	68	992	2083	273	6
31	2	-	10	480	3645	5654	813	42
32	2	5	72	657	3036	4167	757	27
33	2	-	2	296	3025	5708	1017	32
34	8	-	-	22	421	1483	358	4
35	80	-	2	26	26	123	37	2
36	24	-	-	16	52	95	-	-

Table 4: Minimal defining set classes of the 2-$(10, 4, 2)$ designs

		Counts				
Design	|Aut|	5	6	7	8	9
1	48	-	3	13	-	-
2	24	1	59	-	-	-
3	720	-	-	-	4	1

Table 5: Minimal defining sets of 2-$(8, 4, 3)$ designs #1 and #3

#1	6	7	8	#3	6
0123	`XXXX-XX`	`X-X`	`--`	0123	`XXXXXX---XX-XX--XXX-------X`
0124	`XXXXXXX`	`XXX`	`XX`	0124	`XXX-XXXX-XXX----XXXXX-XXX--`
0156	`XX-----`	`XXX`	`X-`	0156	`XX-XX-X-XX-X-XX--X-XXXX-X--`
0257	`X-XXX--`	`---`	`XX`	0257	`X-XX-X-XX-X-X--XX-X-XX-X-XX`
0345	`X--X---`	`X--`	`XX`	0347	`-XXX--XXX--XX-XXX--XXX---XX`
0367	`-XX-XXX`	`-X-`	`XX`	0356	`X---XXXXXXXX-XXX-XXX--XXX--`
0467	`-XX-XXX`	`XXX`	`--`	0467	`-XXXXXXXX---XXXX-----XXX-X-`
1267	`---X---`	`-X-`	`-X`	1267	`---------XXXXXXXXXXX----XXX`
1346	`XXXXX--`	`XXX`	`XX`	1345	`XXXXXXXXXXXXXXXX----XXXXXXX`
1357	`----XX-`	`XXX`	`XX`	1367	`----------------XXXXXXXXXX-`
1457	`-----XX`	`---`	`--`	1457	`--------------------------X`
2347	`-------`	`---`	`--`	2346	`---------------------------`
2356	`------X`	`--X`	`XX`	2357	`---------------------------`
2456	`-------`	`---`	`--`	2456	`---------------------------`

Table 6: Minimal defining sets of 2-$(8, 4, 3)$ designs #2 and #4

#2	6	7	#4	6	8
0123	`XX-XXXXXXXX--------XX-XXX-X-XXX-`	`--`	0123	`-`	`X`
0124	`XXXXXXXXXXXXXXX-X--XXXXXXX-XXXX-`	`XX`	0145	`-`	`X`
0156	`XXXX-X-X-X-XXX-XXX-XXX--XX-XX--X`	`-X`	0167	`-`	`-`
0257	`XXX-X-X-X-X-X-XX--X-------X----X`	`XX`	0246	`-`	`X`
0345	`X-XXX--XX--X-------X-X---X------`	`-X`	0257	`X`	`X`
0367	`-XX--XX--XXXXXXXXXX-XXXXXXXXXXX-`	`XX`	0347	`X`	`X`
0467	`---XXXX------XXX-XX---X---X--X--`	`--`	0356	`X`	`-`
1267	`-------XXXXX----XXX----X------X-`	`X-`	1247	`X`	`X`
1347	`------------XXXXXXXXXXXX---X---X`	`X-`	1256	`X`	`X`
1356	`XXXXXXXXXXXXXXXXXXX-----XXXX---X`	`X-`	1346	`X`	`X`
1457	`-------------------XXXXXXXXXXXXX`	`-X`	1357	`-`	`-`
2346	`----------------------------XXXX`	`XX`	2345	`-`	`-`
2357	`--------------------------------`	`--`	2367	`-`	`-`
2456	`--------------------------------`	`--`	4567	`-`	`-`

Table 7: Minimal defining sets of 2-$(10, 4, 2)$ designs #1 and #3

#1	6	7	#3	8	9
0123	XXX	XXXX-XXX-XX-X	0123	XX--	X
0145	-X-	XX--X-XXXX-XX	0145	X-XX	X
0246	XXX	XXXXX-X-X--XX	0267	XXXX	-
0378	X--	XXXXXX-XXXXX-	0389	XXXX	X
0579	X-X	X--X-XXXX-XX-	0468	XXX-	X
0689	-X-	-XX-XX---XX-X	0579	---X	X
1278	---	XXXXXXXXXXX--	1289	XXXX	X
1369	XXX	--XXXX-----XX	1367	-XXX	X
1479	XX-	XXX---XXX----	1479	XXX-	-
1568	--X	---XXX---XXXX	1568	---X	X
2359	--X	------XXXXXXX	2345	----	-
2489	---	-------------	2478	XXXX	X
2567	---	-------------	2569	----	-
3458	---	-------------	3469	----	-
3467	---	-------------	3578	----	-

Table 8: Minimal defining sets of 2-$(10, 4, 2)$ design #2

#2	5	6
0123	X	XXXX-X-XX-XXXX-XX-XX-X-XX-X-X---XXX-----XX-XXX-X-XX-XX-XX--
0145	X	XXX-X-XX-XX-X-XX-XX-X-XX-X-X-XXXXXXXXXXX--XXX-X-XX-XX-XX-XX
0246	X	XXXXXXX---XXXXXXXX---XX---XXXX------X------XXXXXX---XXX---X
0378	-	XX-XXXX-XXXX-XX-XX-XXXX-XXXX--X-XX-XXXXXX-XX-XXXX-XX-XX-XXX
0579	-	X-XXX--XXX-XXXX---XXX-------XX-XX-XX-X-XXX--XXX------------
0689	-	-XX--XXXXX-----XXXXXX--XXX----XX-XXX--X--XX------XXX---XXX-
1279	X	XXXXXXXXXXXX---------XXXXXXXXXXXXXXXXXX-XX-XXXXXXXXX------X
1369	X	---XXXXXXX--XXXXXXXXXXXXXXXXXXXX----XXXX--X----XXXXXXXXXXXX
1478	-	----------XXXXXXXXXXXXXXXX-------------X------------XXXXXX-
1568	-	--------------------------XXXXXX--------XXX----------------
2358	-	--------------------------------XXXXXXXXXXX----------------
2489	-	---XXXXXXXXXXXXXXX-
2567	-	--X
3459	-	---
3467	-	---

References

[1] D.R. Breach, The 2-$(9,4,3)$ and 3-$(10,5,3)$ designs, *Journal of Combinatorial Theory (Series A)* 27 (1979), 50–73.

[2] Catherine M. Delaney, *Computational aspects of defining sets of combinatorial designs*, MSc Thesis, Department of Mathematics, The University of Queensland, 1995.

[3] P. Gibbons, *Computing techniques for the construction and analysis of block designs*, Technical Report 92/76, Department of Computer Science, University of Toronto, 1976.

[4] Rebecca A.H. Gower, Minimal defining sets in a family of Steiner triple systems, *Australasian Journal of Combinatorics* 8 (1993), 55–73.

[5] Brenton D. Gray, Rudolf Mathon, Tony Moran and Anne Penfold Street, The spectrum of minimal defining sets of some Steiner systems, *Discrete Mathematics* 261 (2003), 277–284.

[6] Brenton D. Gray and Colin Ramsay, Some results on defining sets of t-designs, *Bulletin of Australian Mathematical Society* 59 (1999), 203–215.

[7] Ken Gray, On the minimum number of blocks defining a design, *Bulletin of Australian Mathematical Society* 41 (1990), 97–112.

[8] Ken Gray, Further results on smallest defining sets of well known designs, *Australasian Journal of Combinatorics* 1 (1990), 91–100.

[9] Ken Gray, Defining sets of single-transposition-free designs, *Utilitas Mathematica* 38 (1990), 97–103.

[10] Catherine S. Greenhill, An algorithm for finding smallest defining sets of t-designs, *Journal of Combinatorial Mathematics and Combinatorial Computing* 14 (1993), 39–60.

[11] Catherine S. Greenhill and Anne Penfold Street, Smallest defining sets of some small t-designs and relations to the Petersen graph, *Utilitas Mathematica* 48 (1995), 5–31.

[12] George Havas, Julie L. Lawrence, Colin Ramsay, Anne Penfold Street and Emine Şule Yazıcı, *Defining Set Spectra for Designs can have Arbitrarily Large Gaps; Supplementary Materials* (2007). Available from `http://www.itee.uq.edu.au/~havas/defsetspectra`

[13] Abdollah Khodkar, Smallest defining sets for the 36 non-isomorphic twofold triple systems of order 9, *Journal of Combinatorial Mathematics and Combinatorial Computing* 17 (1995), 209–215.

[14] Julie L. Lawrence, Computational techniques for finding smallest defining sets of some discrete structures, talk presented at 23rd ACCMCC, Brisbane, 1998.

[15] Rudolf Mathon and Alexander Rosa, A census of Mendelsohn triple systems of order nine, *Ars Combinatoria* 4 (1977), 309–315.

[16] Brendan D. McKay, *nauty* user's guide (version 2.2), Department of Computer Science, Australian National University, 2004. Available at `http://cs.anu.edu.au/~bdm/nauty`.

[17] Tony Moran, Smallest defining sets for 2-$(9, 4, 3)$ and 3-$(10, 5, 3)$ designs, *Australasian Journal of Combinatorics* 10 (1994), 265–288. Corrigendum, *ibid*. 14 (1996), 311–312.

[18] Elizabeth J. Morgan, Some small quasi-multiple designs, *Ars Combinatoria* 3 (1977), 233–250.

[19] Harikinkar Nandi, Enumeration of non-isomorphic solutions of balanced incomplete block designs, *Sankhya* 7 (1946), 305–312.

[20] Harikinkar Nandi, A further note on non-isomorphic solutions of incomplete block designs, *Sankhya* 7 (1945), 313–316.

[21] Colin Ramsay, An algorithm for completing partials, with an application to smallest defining sets of the $STS(15)$, *Utilitas Mathematica* 52 (1997), 205–221.

[22] Colin Ramsay, An algorithm for enumerating trades in designs, with an application to defining sets, *Journal of Combinatorial Mathematics and Combinatorial Computing* 24 (1997), 3–31.

[23] R.G. Stanton, R.C. Mullin and J.A. Bate, Isomorphism classes of a set of prime $BIBD$ parameters, *Ars Combinatoria* 2 (1976), 251–264.