Contemporary Mathematics
Volume 111, 1990

Constructing 6-(14,7,4) Designs

DONALD L. KREHER AND STANISŁAW P. RADZISZOWSKI

ABSTRACT. A summary of the algebraic and computational techniques used in the construction of two non-isomorphic simple 6-(14,7,4) designs and four non-isomorphic simple 5-(13,6,4) designs is presented. With the exception of the 6-(33,8,36) designs discovered by Magliveras and Leavitt, and the 6-(20,9,112) designs discovered by Kramer, Leavitt and Magliveras, this is the only other small parameter situation in which a simple 6-design is known to exist.

1. Introduction

The results and ideas in this paper are not new but instead are spread over three of our papers [**KR1**], [**KR2**], [**KR4**]. Thus the authors were at first reluctant to enunciate them again. However, due to the encouragement of colleagues and the apparent importance of the results, this paper was presented and received favorably at the 307th AMS meeting held in Lincoln, Nebraska. This is the only paper in which the entire details of our method appear.

2. Notation and background

The construction of the 6-(14,7,4) designs had three essential components.

I: Incidence Matrices
II: Basis Reduction
III: Extension

Each of these components posed difficult and interesting problems both computational and mathematical. They will be discussed in the sections that follow. First recall that a *t-design,* or $t-(v, k, \lambda)$ *design* is a pair $(X, \mathscr{B})$ with a *v-set X of points* and a family $\mathscr{B}$ of k-subsets of X called *blocks* such that any t points are contained in exactly λ blocks. A $t-(v, k, \lambda)$ design $(X, \mathscr{B})$ is *simple* if no block in $\mathscr{B}$ is repeated. A group $G \leq Sym(X)$ is

1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 51E05,68R05.
Research supported by National Science Foundation grant No. CCR-8711229.
This paper is in final form and no version of it will be submitted for publication elsewhere.

0271-4132/90 $1.00 + $.25 per page

an automorphism group of a $t-(v, k, \lambda)$ design $(X, \mathscr{B})$ if every $g \in G$ preserves $\mathscr{B}$. For example a $2-(7, 3, 1)$ design $(X, \mathscr{B})$ is given by $X = \{1,2,3,4,5,6,7\}$ and $\mathscr{B} = \{124,235,346,457,156,267,137\}$. The points on the 6 lines and 1 circle in Figure 1 are the 7 blocks of this design. Thus easy observation shows that it has $G =< (1\ 4\ 5)(2\ 7\ 6), (2\ 6)(4\ 5) > \simeq S_3$ as an automorphism group.

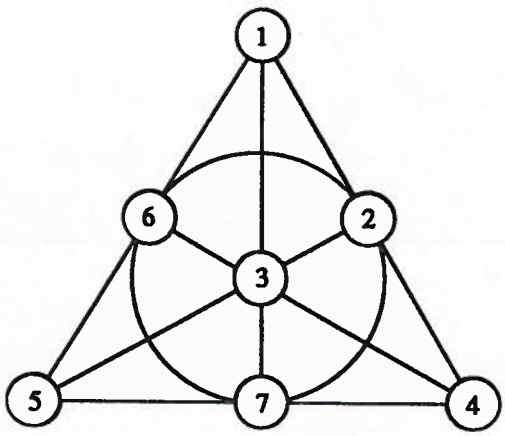

FIGURE 1. The $2-(7, 3, 1)$ design.

The set of *all* automorphisms of a $t-(v, k, \lambda)$ design $(X, \mathscr{B})$ is said to be the *full automorphism group* and is denoted by $Aut(\mathscr{B})$. Indeed, as it is well known, for the 2-(7,3,1) design given above $Aut(\mathscr{B})$ is isomorphic to $\mathrm{PSL}_2(7)$ and thus it has in fact 168 automorphisms.

3. Incidence matrices

Incidence matrices have been investigated by a number of researchers but probably their intimate connection to $t-(v, k, \lambda)$ designs with a given automorphism group was first given by Earl Kramer and Dale Mesner in 1976 [**KM**]. Their observation was:

A $t-(v, k, \lambda)$ design exists with $G \leq Sym(X)$ as an automorphism group if and only if there is an integer solution U to the matrix equation

$$A_{tk}U = \lambda \cdot J_{N_t} \tag{1}$$

where:

(a) *The N_t rows of A_{tk} are indexed by $\Delta_1, \Delta_2, \ldots, \Delta_{N_t}$ the G-orbits of t-subsets of X;*

(b) *The N_k columns of A_{tk} are indexed by $\Gamma_1, \Gamma_2, \ldots, \Gamma_{N_k}$ the G-orbits of k-subsets of X;*

(c) $A_{tk}[\Delta_i, \Gamma_j] = |\{K \in \Gamma_j : K \supseteq T, T \in \Delta_i \text{ fixed}\}|$;

(d) $J_{N_t} = [1, 1, 1, \ldots, 1]^T$.

We call the matrix A_{tk} defined above an incidence matr x, and when it is important to keep track of the automorphism group we write $A_{tk}(G|X)$ for

A_{tk} so that no confusion arises. An example appears in Figure 2. A solution to equation (1) in this case is $U = [0, 1, 0, 0, 0, 0, 1, 0, 0, 1]^T$ and the correspondingly chosen 3-subsets form a 2-(7,3,1).

	123		125		127					
	347		147		467					
	136		146		167					
	356	124	456	126	256	134	137		236	
	357	457	157	247	257	345	346		237	
	234	156	245	567	246	135	235	145	367	267
{12 47 16 56 57 24}	1	1	1	1	1	0	0	0	0	0
{13 34 35}	2	0	0	0	0	2	1	0	0	0
{14 45 15}	0	1	2	0	0	1	0	1	0	0
{17 46 25}	0	0	2	0	2	0	1	0	0	0
{23 37 36}	2	0	0	0	0	0	1	0	2	0
{26 27 67}	0	0	0	1	2	0	0	0	1	1
		↑					↑			↑

FIGURE 2. $A_{23}(G \mid X)$, $G =< (1\ 4\ 5)(2\ 7\ 6), (2\ 6)(4\ 5) >$

This observation led directly to the discovery of many previously unknown designs and we mention a few.

1975: Two 5-(17,8,80) designs. The first examples of 5-designs on an odd number of points, Kramer [**K1**].

1982: A large number of 5-(33,6,12), 5-(33,7,42) and 5-(33,7,126) designs. The second, third and fourth sets of examples of 5-designs on an odd number of points, Magliveras and Leavitt [**ML**].

1982: Thirteen 6-(33,8,36) designs. The first examples of 6-designs, Magliveras and Leavitt [**ML**].

1984: Two 6-(20,9,112) designs. The second set of examples of 6-designs, Kramer, Leavitt and Magliveras [**KLM**].

1986: Four 5-(13,6,4) designs. The fifth set of examples of 5-designs on an odd number of points, Kreher and Radziszowski [**KR2**].

1986: Two 6-(14,7,4) designs. The third set of examples of 6-designs, and the smallest possible 6-designs that can exist. Kreher and Radziszowski [**KR2**].

We also point out that an initial investigation of the algebraic properties of the A_{tk} matrices was done in [**K2, K3**] where some new combinatorial identities were found.

Now given this mathematical motivation our first computational problem is:

How can the matrix A_{tk} be computed efficiently?

3.1. Computing A_{tk}. If $G \leq Sym(X)$, then for any integer t, $0 \leq t \leq v$ the number of orbits of t-subsets can be computed by using the well known Cauchy-Frobenius-Burnside Lemma

$$N_t = \sum_{g \in G} \chi(g)$$

where $\chi(g)$ is the number of t-subsets fixed by g. It is easy to determine the value of $\chi(g)$ from the cycle representation of g on the point set X. If $G \leq Sym(X)$ and $0 < t < k < v$, where $v = |X|$, then given orbit representatives $\{T_i : 1 \leq i \leq N_t\}$ of t-subsets and orbit representatives $\{K_i : 1 \leq i \leq N_k\}$ of k-subsets it is easy to see that the A_{tk} matrix of G acting on X can be efficiently generated by making one pass over the group elements. For example algorithm Mat computes A_{tk} in time $O(|G| \cdot N_t \cdot N_k)$.

Algorithm Mat
Let A be an N_t by N_k array with each entry set to 0.
Let $stab$ be a 1-dimensional array of N_k entries all set to 0.
for each (g, j) where $g \in G$ and $1 \leq j \leq N_k$ do
  if $K_j = K_j^g$ then $stab[j] = stab[j] + 1$;
  for each i, $1 \leq i \leq N_t$ do
    if $T_i \subseteq K_j^g$ then $A[i, j] = A[i, j] + 1$;
for each (i, j) where $1 \leq i \leq N_t$ and $1 \leq j \leq N_k$ do
  $A[i, j] = A[i, j]/stab[j]$.

Thus the problem of constructing an A_{tk} matrix is reduced to being able to compute a complete list of orbit representatives $Reps(t) = \{T_i : 1 \leq i \leq N_t\}$ of t-subsets and a complete list of orbit representatives $Reps(k) = \{K_i : 1 \leq i \leq N_k\}$ of k-subsets.

In our next algorithm "$\leq$" is a total linear ordering on the set of all subsets of X. In particular if we take $X = \{0, 1, 2, \ldots, v-1\}$, then $code : P(X) \to \mathbb{Z}$ given by

$$code(S) = \sum_{i \in S} 2^i$$

gives such a linear ordering. That is

$$S \leq T \text{ if and only if } code(S) \leq code(T).$$

This is particularly effective when $|X| \leq$ "the machine word size" (e.g. 32) since one can take advantage of bit operations for doing set operations. If for some t a complete list of orbit representatives $Reps(t)$ of t-subsets is known we define $XReps(t+1)$ to be $\{S \cup \{x\} : S \in Reps(t) \text{ and } x \in X - S\}$.

It is easy to see for any complete list of orbit representatives, $Reps(t+1)$, of $(t+1)$-sets that for all $S \in Reps(t+1)$ there is a $g \in G$ such that S^g, the image of S under g, is an element of $XReps(t+1)$. This observation leads to algorithm **Reps**.

Let L be an empty list to which the following two operations apply: INSERT S, which means insert S on list L and DELETE S, which means delete S from list L.

Algorithm Reps
for each $S \in Reps(t)$ do
  for each $x \in X - S$ do
    INSERT $\{S \cup \{x\}\}$;
$n \leftarrow N_t \cdot (v - t)$;
repeat
  for each S *on* L do
    for each $g \in G$ do
      $K \leftarrow S^g$;
      if $K < S$ then
        DELETE S
        if K is on L then
          $n \leftarrow n - 1$
        else
          INSERT K
        $S \leftarrow K$
until $(n = N_{t+1})$;
$Reps(t+1) = \{S : S$ is on list L$\}$.

This algorithm is the (asymptotically) best algorithm we know of for computing $Reps(t+1)$ from $Reps(t)$. It is easy to modify **Reps** so that the output, $Reps(t+1)$, has the property that whenever $S \in Reps(t+1)$ and $g \in G$, then $S \leq S^g$. This canonical form for $Reps(t+1)$ is often useful in other applications. Since $Reps(0) = \{\varnothing\}$ it is easy to construct from algorithm **Reps** an algorithm to compute a complete list of orbit representatives of t-sets for all $0 \leq t \leq k$. An algorithm similar in spirit but different in implementation is used by Magliveras [**M**]. A different algorithm has also been constructed by Leo Chouinard [**C**]. Although his algorithm uses ideas similar to ours he makes a clever use of data structures to compute the orbit representatives and the resulting algorithm appears to run faster in practice. Its asymptotic running time has not yet been determined. A probabilistic algorithm has also been successfully used by Kramer, Leavitt and Magliveras [**KLM, ML**].

Now that an efficient method has been developed to obtain the matrix A_{tk} we ask the second and more difficult question.

How can equation (1) *be effectively solved?*

4. Solving equation (1)

Logically our approach to solving the integer linear equation (1) does not rely on the fact that the matrix A_{tk} is special, however the performance of our algorithm on general matrices has not been investigated. The related general decision problem is known to be NP-complete and thus we do not present an efficient deterministic algorithm. Instead we use the powerful tool of basis reduction [LLL] with some heuristics and do not guarantee that a solution, if it exists, will always be found. On the other hand we know of no situation for $t-(v, k, \lambda)$ designs and automorphism group G with $N_k < 200$ in which a solution is known to exist but our algorithm is unable to find it.

Let X be a v-set, $G \leq Sym(X)$ and consider the $(m+n)$ by $(n+1)$ matrix B below:

$$B = \begin{bmatrix} I_n & 0 \\ A_{tk} & -\lambda J_m \end{bmatrix}, \tag{2}$$

where A_{tk} is the m by n Kramer-Mesner matrix described in (1) I_n is the n by n identity matrix, and $J_m = [1, 1, 1, \ldots 1]^T$. Let L be the $n+1$ dimensional integer lattice spanned by the columns of B. That is

$$L = \{R \in \mathbb{Z}^{m+n} : R = B \cdot S, \text{ for some } S \in \mathbb{Z}^{n+1}\}.$$

Let E_m be the m-dimensional zero vector. Then the following proposition is clear.

PROPOSITION 1. *$A_{tk}U = d \cdot \lambda J_m$ for some integer d if and only if $[U, E_m]^T \in L$.*

Thus to find a (0,1)-solution U to $A_{tk}U = \lambda J_m$ we need only look for a linear combination $U = [U, E_m]^T$ of the columns of B such that U is a (0,1)-vector. If $U \neq J_m$, then we will have found a nontrivial $t-(v, k, d \cdot \lambda)$ design for some positive integer d. Note that since the complement of a design is a design, then we may assume $\|U\|^2 \leq n/2$. That is, U is a "short" vector in L. Our algorithm will try to find for L a new basis all of whose vectors are as short as we can make them. It has been our experience that the vector U tends to appear as one of the vectors in such a new basis.

4.1. Basis reduction. Let n be a positive integer. A subset L of the r-dimensional real vector space R^r, $r \geq n$ is called a *lattice* if and only if there is a basis $B = \{b_1, b_2, \ldots, b_n\}$ of an n-dimensional subspace of R^r such that every member of L is an integer linear combination of the vectors in B. Recall that given a basis $B = \{b_1, b_2, \ldots, b_n\}$ of an n-dimensional subspace of R^r, an orthogonal basis $B^* = \{b_1^*, b_2^*, \ldots, b_n^*\}$ of it may be obtained inductively via the Gram-Schmidt process of orthogonalization as

follows:

$$b_i^* = b_i - \sum_{j=1}^{i-1} \mu_{ij} b_j^*, \quad \text{for } 1 \le i \le n, \tag{3}$$

$$\mu_{ij} = (b_i, b_j^*)/(b_j^*, b_j^*), \quad \text{for } 1 \le j \le i \le n, \tag{4}$$

where $(\cdot, \cdot)$ denotes the ordinary inner product on R^r. An ordered basis $B = [b_1, b_2, \ldots, b_n]$ for a lattice L will be said to be *y-reduced* (*or reduced*) if the following two conditions hold:

(i) $|\mu_{ij}| \le 1/2$ for $1 \le j < i \le n$,

(ii) $\|b_i^* + \mu_{i,i-1} b_{i-1}^*\|^2 \ge y \cdot \|b_{i-1}^*\|^2$ for $1 < i \le n$,

where y, $\frac{1}{4} < y < 1$ is a constant and $\|\cdot\|$ denotes ordinary Euclidean length. Lenstra et al. [LLL] describe an algorithm which, when presented with y, $\frac{1}{4} < y < 1$ and an ordered basis $B = [b_1, b_2, \ldots, b_n]$ for a lattice L as input, produces a reduced ordered basis $B' = [b_1', b_2', \ldots, b_n']$ as output. The $\mathbf{L}^3$ algorithm consists of applying a finite number of two kinds of linear transformations. These are:

T1: Interchange vectors b_i and b_{i-1} if $\|b_i^* + \mu_{ii-1} b_{i-1}^*\|^2 \ge y \|b_{i-1}^*\|^2$ does not hold, for some $1 < i \le n$, and the global constant $y \in (\frac{1}{4}, 1)$.

T2: Replace b_i by $b_i - rb_j$, where $r = round(\mu_{ij})$ is the nearest integer to μ_{ij}, and $|\mu_{ij}| > \frac{1}{2}$, for some $1 \le j < i \le n$.

The efficient implementation of the sequence of transformations T1 and T2 relies mainly on the fact that old values of μ_{ij} and $\|b_i^*\|^2$ can be easily updated after each transformation without using the full process of orthogonalization. The $\mathbf{L}^3$ algorithm performs the transformations T1 and T2 using a strategy somewhat resembling the bubble-sort. However as H.W. Lenstra [LLL] remarks, *any* sequence of these transformations will lead to the reduced basis.

The $\mathbf{L}^3$ algorithm terminates when neither T1 nor T2 can be applied and such a situation implies that conditions (i) and (ii) are satisfied. The resulting reduced basis B' is an integer approximation to the basis B^* defined by the Gram-Schmidt orthogonalization process and as a consequence contains a "short" vector, as can be seen in the following proposition of Lenstra et al. [LLL, Proposition 1.11]:

PROPOSITION 2. *Let* $B' = [b_1', b_2', \ldots, b_n']$ *be a reduced basis of a lattice* L. *Then*

$$\|b_1'\|^2 \le 2^{n-1} \cdot \min\{\|b\|^2 : b \in L \text{ and } b \ne 0\}.$$

They also give the following polynomial worst-case running time for its performance [LLL, Proposition 1.26].

PROPOSITION 3. *Let* $B = [b_1, b_2, \ldots, b_n]$ *be an ordered basis for an integer lattice* L *and let* Max *be an integer such that* $\|b_i\|^2 \leq \text{Max}$ *for* $1 \leq i \leq n$. *Then the* $\mathbf{L}^3$ *algorithm produces a reduced basis* $B' = [b'_1, b'_2, \ldots, b'_n]$ *for* L *using at most* $O(n^4 \log_2 \text{Max})$ *arithmetic operations, and the integers on which these operations are performed have length at most* $O(n \log -2 \text{Max})$.

In summary, the effect of the $\mathbf{L}^3$ algorithm is such that when given a basis B of an n-dimensional lattice $L \subseteq \mathscr{Z}^r$ it produces a *reduced* basis B' of L, and

(i) $\mathbf{L}^3$ uses at most $O(n^4 \log \text{Max})$ arithmetic operations,
(ii) B' is *almost* orthogonal (integer approximation to Gram-Schmidt orthogonal basis),
(iii) B' contain a "short" vector.

Furthermore, we point out that although it is only proven [**LLL**] that B' contains a vector not longer than $2^{(n-1)/2} \cdot s$, where s is the length of a shortest nonzero vector in L, in practice the $\mathbf{L}^3$ algorithm finds much shorter vectors [**LO**].

When the number of rows in the A_{tk} matrix is $m = 1$, then (1) reduces to the knapsack or subset-sum problem. The application of the $\mathbf{L}^3$ algorithm to solve the subset sum problem was first studied in 1985 by J. C. Lagarias and A. M. Odlyzko [**LO**]. Our improvements in this direction appear in [**KR3**].

When the number of rows in the A_{tk} matrix is greater than 1, then our experience has been that $\mathbf{L}^3$ by itself often doesn't find t-designs. Thus further reduction methods are necessary.

4.2. Weight reduction. If B is the $(n+1)$-dimensional reduced basis produced by the $\mathbf{L}^3$ algorithm applied to (2), then there often exist, as extensive experiments showed, pairs of indices i and j, $1 \leq i, j \leq n+1$, $i \neq j$, and $\epsilon \in \{+1, -1\}$ such that

$$\text{if } v = b_i + \epsilon b_j, \text{ then } \|v\| < \max\{\|b_i\|, \|b_j\|\}. \tag{5}$$

A pair (i, j), $i \neq j$, satisfies the last condition for some $\epsilon \in \{+1, -1\}$ if and only if $\min\{\|b_i\|^2, \|b_j\|^2\} < 2 \cdot |(b_i, b_j)|$. In such a case we take ϵ to have a different sign from (b_i, b_j) and substitute v for the longer of b_i and b_j, obtaining a new basis with decreased *total weight*

$$w(B) = \sum_{p=1}^{n+1} \|b_p\|^2.$$

To facilitate the process of finding successive pairs of indices (i, j) satisfying (5) and decreasing $w(B)$ the algorithm keeps an array containing (b_i, b_j) and $\|b_i\|^2$ for $1 \leq i, j \leq n+1$. Whenever a pair of indices (i, j) is found satisfying (5) leading to substitution of some b_i by v these arrays

must be updated. It is not necessary however to recalculate $\|v\|^2$ and (v, b_k) for $k \neq i$ from the definitions. Instead we keep track of the integers $\|b_i\|^2$ and $inn_{ij} = (b_i, b_j)$, for $1 \leq j < i < n+1$. These are then sufficient since we have the formulas:

$$\|v\|^2 = \|b_i\|^2 + \|b_j\|^2 - 2|inn_{ij}|, \text{ and}$$

$$(v, b_k) = inn_{ik} + \epsilon \cdot inn_{jk}, \text{ for } 1 \leq k \leq n+1, \ k \neq i \text{ and } k \neq j.$$

A simple algorithm for finding all such pairs can be designed and implemented in time $O(n^2)$ for each reduction, producing as output a basis with smaller weight. This algorithm, let us call it *Weight-Reduction*, is a useful complement to the $\mathbf{L}^3$ algorithm. When used as follows, the algorithms $\mathbf{L}^3$ and *Weight-Reduction* jointly tend to produce much shorter vectors than using $\mathbf{L}^3$ or *Weight-Reduction* alone:

$B \leftarrow \mathbf{L}^3(B)$;
repeat
    Weight-Reduction;
    sort basis with respect to $\|b_i\|^2$;
    $B \leftarrow \mathbf{L}^3(B)$;
until ($w(B)$ does not decrease);
Weight-Reduction.

The $\mathbf{L}^3$ algorithm can remove the vector b_1 from the basis B only by replacing it with a shorter vector, since for $i = 2$ if the transformation T1 can be applied then $\|b_i\|^2 < \|b_{i-1}\|^2$ (note that this is not true when $i > 2$). Hence sorting the basis with respect to $\|b_i\|^2$ guarantees that the shortest vector in the basis B will not disappear in the next iteration, unless a new shorter vector is found.

Following the above approach one can try in general to find a k-tuple of distinct vectors $b_{i_1}, \ldots, b_{i_k}$, for some $k \geq 2$, in the basis B, such that the vector

$$v = \sum_{p=1}^{k} \epsilon_p b_{i_p}, \text{ for some choice of } \epsilon_p = \pm 1, \ 1 \leq p \leq k,$$

is shorter than b_{i_k}, where b_{i_k} is the longest vector in the k-tuple. In the latter case the weight of basis B can be decreased by substituting v for b_{i_k} Note that $\|v\| < \|b_{i_k}\|$ if and only if

$$\|v\|^2 = \sum_{j=1}^{k} \|b_{i_j}\|^2 + \sum_{h \neq j} \epsilon_{i_h} \epsilon_{i_j} (b_{i_h}, b_{i_j}) < \|b_{i_k}\|^2, \tag{6}$$

and a necessary condition for (6) is

$$\sum_{j=1}^{k-1} \|b_{i_j}\|^2 < \sum_{h \neq j} \left| (b_{i_h}, b_{i_j}) \right|.$$

Consequently, our approach is to search for such k-tuples of vectors by considering the complete graph whose vertices are the basis vectors b_i and whose edges are labeled by edge weights $|(b_i, b_j)|$. The endpoints of edges with large weight are "less" orthogonal, and hence they are good candidates for the desired k-tuple. We can try to construct it by finding subgraphs with large edge weight.

Obviously, the complete analysis of all subgraphs would be too expensive. However, we are satisfied with a heuristic search for just a few of relatively small size. As before they are used to decrease the weight of basis B. This technique leads to a generalization of the *Weight-Reduction* algorithm and improves further the behavior of the $\mathbf{L}^3$ algorithm. We have implemented this strategy for $k = 3$ and $k = 4$.

4.3. Size reduction. Recall that $[U, E_m]^T \in \mathbf{L}$ if and only if there exist integers $a_1, a_2, \ldots, a_{n+1}$ such that

$$\begin{bmatrix} U \\ E_m \end{bmatrix} = B \begin{bmatrix} a_1 \\ \vdots \\ a_{n+1} \end{bmatrix}.$$

Whence, it follows that:

(*) *If there is exactly one* j *such that* $b_{hj} \neq 0$ *for some* h, $n < h \leq n+m$, *then* $a_j = 0$.

In this case we let B' be B with row h and column j removed and L' be the lattice spanned by B'. Then the $(n + m - 1)$-dimensional vector $[U, E_{m-1}]^T \in L'$ if and only if the $(n+m)$-dimensional vector $[U, E_m]^T \in L$.

To achieve situation (*) for row h, $n < h \leq n+m$, we perform the following two operations

(1) Multiply row h by $c = \max_i \|b_i\|^2$.

(2) Apply *Weight-Reduction* and/or $\mathbf{L}^3$.

This almost always produces such a situation. If this procedure is successfully iterated for each h, $h = n + m, n + m - 1, \ldots, n + 1$, then the resulting basis B' will consist of $n - m + 1$, n-dimensional vectors. Furthermore

$$U \in L' \text{ if and only if } A_{tk}U = d \cdot \lambda J_m,$$

for some integer d (see Proposition 1). Thus the result of these iterations, let us call them collectively *Size-Reduction,* is a basis of short vectors for the integer solution space to the matrix equation $A_{tk}U = d \cdot \lambda J_1$. Consequently, to discover a $t-(v, k, d \cdot \lambda)$ design we need only search the lattice spanned by B' for a (0,1)-vector. Finally, our complete algorithm is given in Figure

3.

Algorithm MSV (Matrix Short Vector)
input basis B of the form in (2);
$B \leftarrow \mathbf{L}^3(B)$;
$B \leftarrow$ *Size-Reduction*, (B);
repeat
Weight-Reduction;
sort basis with respect to $\|b_i\|^2$;
$B \leftarrow \mathbf{L}^3(B)$;
until $(weight(B) = \sum \|b_i\|^2$ does not decrease);
Weight-Reduction.
Check for solution after each *Weight-Reduction* and $\mathbf{L}^3$.

FIGURE 3. The MSV algorithm.

5. Extension

If $(X, \mathscr{B})$ is a $t-(v, k, \lambda)$ design and $x \in X$ then $(\Delta, \mathscr{B}_x)$ where $\Delta = X-\{x\}$ and $\mathscr{B}_x = \{K-\{x\} : x \in K \text{ and } K \in \mathscr{B}\}$ is a $(t-1)-(v-1, k-1, \lambda)$ design and is said to be the *derived design with respect to x*. Similarly, if there is a point $\infty \notin X$ and a $(t+1)-(v+1, k+1, \lambda)$ design $(Y, \mathscr{D})$ whose derived design with respect to $\infty \in Y$ is $(X, \mathscr{B})$, we say that $(Y, \mathscr{D})$ is an *extension of* $(X, \mathscr{B})$ *by* ∞.

Let G be an automorphism group of a $t-(v, k, \lambda)$ design $(X, \mathscr{B})$. Then there is a (0,1)-vector U such that $A_{tk}(G \mid X) \cdot U = \lambda J_{N_t}$. Further suppose that $\infty \notin X$, set $Y = X \cup \{\infty\}$ and let $H \leq Sym(Y)$ be defined by $h \in H$ if $h(\infty) = \infty$ and $h \mid X \in G$. Then it is easy to see, rearranging orbits if necessary, that

$$A_{t+1,k+1}(H \mid Y) = \begin{bmatrix} A_{tk}(G \mid X) & 0 \\ A_{t+1,k}(G \mid X) & A_{t+1,k+1}(G \mid X) \end{bmatrix},$$

where 0 is the matrix of all zeros. Consequently, an $(X, \mathscr{B})$ has an extension to a $(t+1)-(v+1, k+1, \lambda)$ design $(Y, \mathscr{D})$ if and only if there is a (0,1)-vector V such that

$$A_{t+1,k+1}(H \mid Y) \begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} \lambda J_{N_t} \\ \lambda J_{N_{t+1}} \end{bmatrix}.$$

In particular V is a solution to

$$A_{t+1,k+1}(G \mid X) \cdot V = \lambda J_{N_{t+1}} - A_{t+1,k}(G \mid X) \cdot U. \tag{7}$$

Such a solution can be found by using the basis reduction method described in previous sections. However, if t is odd and the parameters of the design satisfy $v = 2k+1$ and $|\mathscr{B}| = \frac{1}{2}\binom{v}{k}$, then by Theorem A of Alltop in [A] a solution is immediate. We would also like to point out that if $k+t+2 = v$,

then by Lemma 10 of [K2] $A_{t+1,k+1}(G \mid X)$ has an inverse and thus a (0,1)-solution to (7), if it exists, would represent a unique extension.

6. Cyclic 5-(13,6,4) designs

If $X = \{0, 1, 2, \ldots, 12\}$ and $G = <(0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12)>$, then G is cyclic and the A_{56} matrix belonging to G has 99 rows and 132 columns. A direct search applied to the reduced basis obtained from the basis reduction algorithm after doing size reduction found that there are exactly two non-isomorphic solutions. A complete description of these designs can be found in [KR2]. Also in the same paper it was established that these two designs both have full automorphism group cyclic of order 13. Furthermore, between them they partition all of the 6-subsets.

If $(X, \mathscr{B})$ is a $t-(v, k, \lambda)$ design then $(X, \bar{\mathscr{B}})$, where $\bar{\mathscr{B}} = \binom{X}{k} - \mathscr{B}$, is a $t-(v, k, \binom{v-t}{k-t} - \lambda)$ design and is called the *complementary design of* $(X, \mathscr{B})$. Using Alltop's Theorem [A] any 5-(13,6,4) design $(X, \mathscr{B})$ can be extended to a 6-(14,7,4) design $(X \cup \{\infty\}, \mathscr{B}' \cup \mathscr{B}'')$ where

$$\mathscr{B}' = \{K \cup \{\infty\} : K \in \mathscr{B}\},$$
$$\mathscr{B}'' = \{X - K : K \in \bar{\mathscr{B}}\}.$$

We note that the complementary design of a 6-(14,7,4) design is also a 6-(14,7,4) design. These two 6-designs each have full automorphism group cyclic of order 13 and they partition all of the 7-subsets.

Using the results in Section 5 it is easy to see that the extension from any 5-(13,6,4) design to a 6-(14,7,4) design is unique. So we have the following theorem.

THEOREM 4. *If* $(X, \mathscr{B})$ *is a* 5-(13,6,4) *design it has a unique extension to a* 6-(14,7,4) *design.*

Recall that if $G \leq Sym(X)$ fixes a subset $\Delta \subseteq X$, then each permutation $g \in G$ induces a permutation g^{Δ} on Δ and we denote the totality of g^{Δ}'s formed, by G^{Δ}. Note that G_x, the stabilizer in G of x, fixes the set $\Delta = X - \{x\}$.

THEOREM 5. *Let* $(X, \mathscr{B})$ *be a* 6-(14,7,4) *design with full automorphism group G. Then for each* $x \in X$, *the derived design with respect to x has full automorphism group* G_x^{Δ} *where* $\Delta = X - \{x\}$.

PROOF. Let $(X, \mathscr{B})$ be a 6-(14,7,4) design. Fix $x \in X$, set $\Delta = X - \{x\}$ and let H be the full automorphism group of the derived design $(\Delta, \mathscr{B}_x)$ of $(X, \mathscr{B})$ with respect to x. It is easy to see that $G_x^{\Delta} \subseteq H$. However, we note by Theorem 4 that $(X, \mathscr{B})$ is the unique extension of $(\Delta, \mathscr{B}_x)$ so it must be obtained by Alltop's construction [A]. That is

$$\mathscr{B} = \{K \cup \{x\} : K \in \mathscr{B}_x\} \cup \{\Delta - K : K \in \bar{\mathscr{B}}_x\}.$$

Observe that H is also the full automorphism group of the complementary design $\bar{B}_x$. Whence, if $\alpha \in H$, then $(\Delta - K)^{\alpha} = \Delta - (K^{\alpha}) \in \mathscr{B}$. Consequently

the extension of α to $\hat{\alpha}: X \to X$ given by $y^{\hat{\alpha}} = y^{\alpha}$ if $y \neq x$ and $x^{\hat{\alpha}} = x$ must preserve $\mathscr{B}$. Thus $H \subseteq G_x^{\Delta}$ and hence the two are equal.

THEOREM 6. *Let* $(X, \mathscr{B})$ *be a* 6-(14,7,4) *design and let* $(X, \bar{\mathscr{B}})$ *be its complementary design. Then for each* $x \in X$ *any isomorphism* $\alpha : (X - \{x\}, \mathscr{B}_x) \to (X - \{x\}, \bar{\mathscr{B}}_x)$ *lifts to an isomorphism* $\hat{\alpha} : (X, \mathscr{B}) \to (X, \bar{\mathscr{B}})$.

PROOF. Fix $x \in X$, set $\Delta = X - \{x\}$ and suppose $\alpha : (\Delta, \mathscr{B}_x) \to (\Delta, \bar{\mathscr{B}}_x)$ is an isomorphism. Define $\hat{\alpha} : X \to X$ by $y^{\hat{\alpha}} = y^{\alpha}$ if $y \neq x$ and $x^{\hat{\alpha}} = x$. Then $\hat{\alpha}$ is a isomorphism from $(X, \mathscr{B})$ to $(X, \bar{\mathscr{B}})$ since by Alltop's construction and Theorem 4 we have

$$\mathscr{B} = \{K \cup \{x\} : K \in \mathscr{B}_x\} \cup \{\Delta - K : K \in \bar{\mathscr{B}}_x\},$$
$$\bar{\mathscr{B}} = \{K \cup \{x\} : K \in \bar{\mathscr{B}}_x\} \cup \{\Delta - K : K \in \mathscr{B}_x\}.$$

A t-design is *rigid* if it has no nontrivial automorphism. The lack of automorphisms make them often difficult to find.

COROLLARY 7. *The* 6*-subsets of a* 13*-set can be partitioned into two nonisomorphic rigid* 5-(13,6,4) *designs.*

PROOF. Let $\Delta = \{0, 1, 2, \ldots, 12\}$, $X = \Delta \cup \{\infty\}$ and consider the two complementary 6-(14,7,4) designs $(X, \mathscr{B})$ and $(X, \bar{\mathscr{B}})$ found in **[KR2]**. These two designs are nonisomorphic with full automorphism group cyclic of order 13 generated by the permutation $\alpha = (0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12)(\infty)$. Now fix $x \in \Delta$ and apply Theorems 5 and 6. The result follows.

This section suggests the following problems.

PROBLEM 1. *Does there exist a way to partition the* 6*-subsets of a* 13*-set into two isomorphic* 5-(13,6,4) *designs?*

PROBLEM 2. *Describe in a compact way the* 6-(14,7,4) *designs found in* **[KR2]** *without listing all of the orbit representatives.*

7. Concluding remarks

The methods described above were also successful in finding some other designs. These designs are listed in Table 1. Other small configurations when needed, even though their parameter situation had been settled, were also found quite easily with this method. We believe that basis reduction and similar tools will become a valuable aid to the combinatorialist and are currently conducting research to enhance the productivity of our algorithms.

TABLE 1

Parameters	Group	Remarks	Reference
$5-(28,6,\lambda)$	$PSL_2(27)$	New designs for all λ, $2 \leq \lambda \leq 21$	[KR4]
$4-(12,6,10)$	C_{11}	New design	[KR3]
$4-(15,5,5)$	C_{13}	New design	[K4]
$3-(20,5,\lambda)$	S_6	New designs for $\lambda \in \{18,28,48,58\}$	[KdeC]
$3-(20,5,\lambda)$	A_6	New designs for $\lambda \in \{24,54\}$	[KdeC]
$3-(20,5,\lambda)$	$H \leq AF(19)$, $\lvert H\rvert = 114$	New designs for $\lambda \in \{12,42,22,52,34,64\}$	[KdeC]

Note added in proof: Several thousand new designs were recently found by Kreher, Chee, DeCaen, Colburn, and Kramer, using the methods described in this paper. (Cf. *Some new simple t-designs*, to appear in the Journal of Combinatorial Mathematics and Combinatorial Computing.)

REFERENCES

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

[C] L. G. Chouinard II, Personal communication..

[K1] E. S. Kramer, *Some t-Designs for $t \geq 4$ and $t = 17, 18$*, Congressus Numerantium XIV, Proceedings of the Sixth Southeastern Conference on Combinatorics, Graph Theory and Computing (1975), 443–460.

[K2] D. L. Kreher, *An Incidence Algebra for t-Designs with Automorphisms*,, Journal of Combinatorial Theory A **42, No. 2** (1986), 239–251.

[K3] ———, *A Generalization of Connor's Inequality to t-Designs with Automorphisms*, Journal of Combinatorial Theory A **50** (1989), 259–268.

[K4] ———, A New 4-(15,5,4) design (to appear).

[KdeC] D. L. Kreher and D. de Caen, *On* 3-(20,5,lambda) *designs*, Rochester Institute of Technology, Computer Science Report: 001-NOV-1988.

[KLM] E. S. Kramer, D. W. Leavitt and S.S. Magliveras, *Construction procedures for t-Designs and the existence of new simple* 6*-designs*, Annals of Discrete Mathematics **26** (1985), 247–274.

[KM] E. S. Kramer and D. Mesner, *t-designs on Hypergraphs*, Discr. Math. **15** (1976), 263–296.

[KR1] D. L. Kreher and S. P. Radziszowski, *Finding Simple t-Designs by Using Basis Reduction*, Congressus Numerantium, Proceedings of the 17-th Southeastern Conference on Combinatorics, Graph Theory and Computing **55** (1986), 235–244.

[KR2] ———, *The Existence of Simple* 6-(14,7,4) *designs*, Journal of Combinatorial Theory Series A, **43 No. 2** (1986), 237–244.

[KR3] ———, *Solving Subset Sum Problems with the* L^3 *Algorithm*, Journal of Combinatorial Mathematics and Combinatorial Computing **3** (1988), 49–63.

[KR4] ———, *New t-designs found by basis reduction*, Congressus Numerantium, Proceedings of the 18-th Southeastern Conference on Combinatorics Graph Theory and Computing **59** (1987), 155–164.

[LLL] A. K. Lenstra, H. W. Lenstra and L. Lovász, *Factoring Polynomials with Rational Coefficients*,, Mathematische Annalen **261** (1982), 515–534.

[LO] J. C. Lagarias and A. M. Odlyzko, *Solving Low-Density Subset Sum Problems*, Journal of the ACM **32, No. 1** (1985), 229–246.

[M] S. S. Magliveras, Private communication..

[ML] S. S. Magliveras and D. W. Leavitt, *Simple* 6-(33,8,36) *Designs from* $P\Gamma L_2(32)$, Computational Group Theory, Proceedings of the London Mathematical Society Symposium on Computational Group Theory, Academic Press, 1984, pp. 337–352.

SCHOOL OF COMPUTER SCIENCE, ROCHESTER INSTITUTE OF TECHNOLOGY, ROCHESTER, NEW YORK 14623

Current address, D. L. Kreher: Department of Mathematics, University of Wyoming, P. O. Box 3036 University Station, Laramie, Wyoming 82071