A search algorithm for line-transitive, point-imprimitive linear spaces

Anton Betten, Gregory Cresp, Alice C. Niemeyer
and Cheryl E. Praeger*

May 3, 2005

Abstract

We present an algorithm that constructs and classifies finite linear spaces admitting a given line-transitive group which leaves invariant a nontrivial point-partition. This algorithm was developed from an algorithm of Nickel and Niemeyer that classified such linear spaces on 729 points with line-size 8, but was never published. It has been applied to complete a classification of such spaces for which the Delandtsheer-Doyen parameters are small.

Key Words: linear spaces, block design, line-transitive, point-imprimitive, combinatorial search algorithms.

AMS subject classification: 05B05, 05B25, 20B25

1 Introduction

A finite *linear space* is an incidence structure $\mathcal{S} = (\mathcal{P}, \mathcal{L})$ where $\mathcal{P}$ is a finite set of points, $\mathcal{L} = \{\lambda_1, \ldots, \lambda_b\}$ is a set of subsets of $\mathcal{P}$ called lines, such that each pair of points lies in a unique line, and each line

*The research for this paper forms part of an Australian Research Council Large Grant. The authors also acknowledge a grant of computing time from the Australian Partnership for Advanced Computing (APAC).

contains at least two points. An automorphism of $\mathcal{S}$ is a permutation of $\mathcal{P}$ which leaves $\mathcal{L}$ invariant, and $\mathcal{S}$ is called *line-transitive* if its automorphism group $\mathrm{Aut}(\mathcal{S})$ is transitive on $\mathcal{L}$. In particular, for a line-transitive linear space $\mathcal{S}$, all lines have the same size k say, and $\mathcal{S}$ is called *non-trivial* if $2 < k < |\mathcal{P}|$. By a theorem of Block [2], each line-transitive subgroup of $\mathrm{Aut}(\mathcal{S})$ is also transitive on $\mathcal{P}$.

The major result that inspired our interest in line-transitive linear spaces, and ultimately this paper, is a theorem of Delandtsheer and Doyen [5] (see Theorem 2.1): if a line-transitive subgroup of $\mathrm{Aut}(\mathcal{S})$ leaves invariant a nontrivial partition of $\mathcal{P}$ (that is to say, if the subgroup acts imprimitively on $\mathcal{P}$), then $|\mathcal{P}| \leq \left(\binom{k}{2} - 1\right)^2$. This upper bound was shown to be sharp precisely when $k = 8$, and there are exactly 467 linear spaces $\mathcal{S}$ that attain the bound (see [9]).

In this paper, we present an algorithm that constructs and classifies line-transitive linear spaces admitting a given line-transitive, point-imprimitive subgroup of automorphisms. These subgroups need not coincide with $\mathrm{Aut}(\mathcal{S})$, the full automorphism group. The algorithm makes full use of the symmetry of the design provided by the subgroup and is a development of the algorithm of Nickel and Niemeyer [8] that was used to construct and classify the designs in [9], but was never published.

Line-transitive, point-imprimitive automorphism groups G of finite linear spaces were investigated further in [6] for the special case where some non-identity element of G fixes each class of the point partition setwise. In this situation, if the 'Delandtsheer-Doyen parameters' are small, then the linear spaces were classified in [6, Theorem 1.4]. The proof of this result required the elimination of one parameter set for which it turned out that no linear spaces existed, namely spaces with 1431 points, line size 11, such that a line-transitive group preserved a point partition into 53 classes of size 27. The algorithm presented in this paper was implemented and applied to this situation, checking by exhaustive search that there were no examples (see Theorem 2.2).

Some background discussion and preparatory results are given in Section 2, and the algorithm is presented and proved in Section 3. The algorithm is essentially a backtrack search involving several additional features. On the one hand, two conditions are introduced (the partial orbit condition and the partial intersection condition) that allow significant reduction of the search tree. In addition, maximum use is made of the given group G of automorphisms to reduce the search further. This involves computation within the normaliser of G in the symmetric group on $\mathcal{P}$. The crucial component of the algorithm is given in Algorithm 1, and this is called recursively in the final algorithm Algorithm 2.

2 Preliminaries

2.1 Delandtsheer-Doyen parameters

A partition $\mathcal{C}$ of a finite set X is a set of pairwise disjoint subsets whose union equals the set X. The subsets $C \in \mathcal{C}$ are called *parts* or *classes* of $\mathcal{C}$ and $\mathcal{C}$ is called *trivial* if either $\mathcal{C}$ consists of only one class or it contains only one-element classes. Otherwise $\mathcal{C}$ is said to be *nontrivial*. Let G be a group that acts on X. Then G is said to leave a partition $\mathcal{C}$ invariant if, for all $g \in G$ and all $C \in \mathcal{C}$, the image C^g is also a class of $\mathcal{C}$. Also G is said to act *imprimitively* on X if G is transitive on X and there exists a nontrivial G-invariant partition of X. Otherwise, G is said to be primitive on X.
Linear spaces admitting a subgroup of automorphisms that is both line-transitive, and point-imprimitive, deserve special attention due to the following result, which shows that the number of points is bounded above by a function of the line size k.

Theorem 2.1 [5] (Delandtsheer-Doyen parameters) *Let $\mathcal{S} = (\mathcal{P}, \mathcal{L})$ be a non-trivial linear space admitting a line-transitive point-imprimitive automorphism group G. Let $\mathcal{C} = \{C_1, \dots, C_d\}$ be a partition of $\mathcal{P}$ with d classes of size c, and assume that $\mathcal{C}$ is invariant under G. Let x be the number of inner pairs of a line λ, that is, pairs of points $\{\alpha, \beta\} \subseteq \lambda \cap C$ for some $C \in \mathcal{C}$. Then there exists another positive* *integer y such that*

$$c = \frac{\binom{k}{2} - x}{y} \quad \textit{and} \quad d = \frac{\binom{k}{2} - y}{x}. \tag{1}$$

We call the pair (x, y) the *Delandtsheer-Doyen parameters* corresponding to $\mathcal{C}$. By [10, Theorem 1.1(a)],

$$x = \frac{\binom{k}{2}(c-1)}{cd-1} \quad \text{and} \quad y = \frac{\binom{k}{2}(d-1)}{cd-1}, \tag{2}$$

and hence the triples (c, d, k) and (x, y, k) mutually determine each other. Moreover, if (c, d, k) corresponds to (x, y, k) then (d, c, k) has (y, x, k) as its mate. We remark further that while the first of the Delandtsheer-Doyen parameters x has a combinatorial interpretation as described in the theorem, no such meaning is known for the second parameter y in general.

Let $\mathcal{S}$ be a non-trivial linear space with line size k, let G be a block-transitive, point-imprimitive group of automorphisms of $\mathcal{S}$, preserving a point-partition $\mathcal{C}$ consisting of d classes of size c, and let (x, y) be the corresponding Delandtsheer-Doyen parameters given by (2). We shall refer to the 5-tuple (x, y, c, d, k) as the *parameter set* for $(\mathcal{S}, G, \mathcal{C})$. We note that the number of points $v = cd$ is determined. Moreover, since a point-transitive group may leave invariant more than one non-trivial point-partition, corresponding to different values for x, y, c, d, there may be different parameter sets for the same linear space $\mathcal{S}$ and line-transitive, point-imprimitive group G.

Let G act on X and let $\mathcal{C}$ be a G-invariant partition of X. The kernel of G on $\mathcal{C}$ is the subgroup $G_{(\mathcal{C})}$ of elements $g \in G$ with $C_i^g = C_i$ for $i = 1, \ldots, d$. We say that $\mathcal{C}$ is *G-normal* if $G_{(\mathcal{C})}$ is transitive on each of the classes of $\mathcal{C}$. The result [6, Theorem 1.4] mentioned in the introduction gives a classification in the case where the Delandtsheer-Doyen parameters are both very small and the partition $\mathcal{C}$ is G-normal. For clarity of the exposition we state this result below.

Theorem 2.2 [6, Theorem 1.4] *Let $\mathcal{S} = (\mathcal{P}, \mathcal{L})$ be a non-trivial linear space admitting a line-transitive automorphism group G that preserves a non-trivial G-normal partition $\mathcal{C}$ of $\mathcal{P}$ such that $(\mathcal{S}, G, \mathcal{C})$ has parameter set (x, y, c, d, k). Suppose further that $x = 1$ and $y \leq 2$. Then either $\mathcal{S}$ is one of the 467 designs with parameter set $(1, 1, 27, 27, 8)$ constructed in* [9], *or $\mathcal{S}$ is one of two designs with parameter set $(1, 2, 7, 13, 6)$ constructed in* [4, 7].

The following exceptional parameter set arose in the proof of this result in [6, Theorem 1.2, and see Lemma 4.2]:

$$(x, y, c, d, k) = (1, 2, 27, 53, 11).$$

It was proved in [6] that a linear space corresponding to this parameter set would admit a line-transitive group of the form $(Z_{53} \times Z_3^3) \cdot Z_{13}$. Each isomorphism type of group with this structure was tested with our algorithm, proving that there were no line-transitive, point-imprimitive linear spaces with this parameter set, and thus completing the proof of Theorem 2.2.

2.2 Base lines and orbit conditions

Let G act on a set of points $\mathcal{P}$. For a nonnegative integer s, we denote the set of s-subsets of $\mathcal{P}$ by $\mathcal{P}^{\{s\}}$. An orbit of G on $\mathcal{P}^{\{s\}}$ is called an *s-orbit*. Thus a 2-orbit is an orbit on pairs. For $S \subseteq \mathcal{P}$, let S^G denote the orbit of S under G, that is, the set of images S^g for all elements g in G. We call S^G the *span* of S under G. Under certain circumstances, the span of a k-subset $\lambda \subseteq \mathcal{P}$ can be taken as the set of lines of a linear space with line size k. In this case, we say that λ is a *base line* for a (line-transitive) linear space. We also say that λ *spans* a linear space. We will characterize base lines shortly. First, we examine some properties of G-orbits on pairs. For two points α and β, we denote by $\lambda(\alpha, \beta)$ the unique line joining α with β. Recall that the number of lines of a linear space is generally called b and equals $\binom{v}{2}/\binom{k}{2}$, where $v = |\mathcal{P}|$.

Lemma 2.3 *Let G be a line-transitive automorphism group of a linear space $\mathcal{S} = (\mathcal{P}, \mathcal{L})$ with line size k and $|\mathcal{L}| = b$. Then the length of every orbit $\mathcal{O}$ of G on $\mathcal{P}^{\{2\}}$ is divisible by b. In particular, the number of orbits of G on pairs is at most $\binom{k}{2}$.*

Proof. For every pair of points $\{\alpha, \beta\}$, $G_{\{\alpha,\beta\}}$ fixes $\lambda(\alpha, \beta)$, and hence $G_{\{\alpha,\beta\}} \leq G_{\lambda(\alpha,\beta)}$. Thus $b = |\lambda(\alpha, \beta)^G| = |G : G_{\lambda(\alpha,\beta)}|$ divides $|G : G_{\{\alpha,\beta\}}| = \left|\{\alpha, \beta\}^G\right|$. In particular $|\mathcal{O}|/b$ is a positive integer, where $\mathcal{O} := \{\alpha, \beta\}^G$. Counting all pairs of points yields the upper bound on the number of those orbits: $\binom{v}{2} = b \cdot \sum_{\mathcal{O}} |\mathcal{O}|/b \geq \left(\binom{v}{2}/\binom{k}{2}\right) \cdot \sum_{\mathcal{O}} 1$, and hence $\binom{k}{2} \geq \sum_{\mathcal{O}} 1$. □

We call a group *feasible* (for a particular linear space parameter set) if the length of every orbit on pairs is divisible by b.

We are going to present a criterion for testing whether the G-span of a k-subset λ forms the set of lines of a linear space. This well-known criterion is in terms of the integers $\mu(\mathcal{O}, \lambda)$ defined by

$$\mu(\mathcal{O}, \lambda) = \left|\mathcal{O} \cap \lambda^{\{2\}}\right| = \left|\left\{\{\gamma, \delta\} \in \mathcal{O} : \{\gamma, \delta\} \subseteq \lambda\right\}\right|, \qquad (3)$$

for $\mathcal{O}$ an orbit of G on $\mathcal{P}^{\{2\}}$. Note that $\mu(\mathcal{O}, \lambda)$ does not depend on the choice of λ within its G-orbit. We are now able to characterize base lines. The conditions (i) and (ii) of the following result are called *the orbit conditions* for a line-transitive linear space.

Lemma 2.4 *(the orbit lemma) Let G be a group acting on a set of v points $\mathcal{P}$. Let k be a positive integer greater than 2 such that $b = \binom{v}{2}/\binom{k}{2}$ is an integer. Then $\lambda \in \mathcal{P}^{\{k\}}$ is a base line of a linear space with point set $\mathcal{P}$ if and only if*

(i) $\left|\lambda^G\right| = b$ *and*

(ii) $\mu(\mathcal{O}, \lambda) = |\mathcal{O}|/b$ *for each G-orbit $\mathcal{O}$ on $\mathcal{P}^{\{2\}}$.*

This result is well-known. It is shown, for example in [3, Proposition 1.3], that $\mu(\mathcal{O},\lambda)/|\mathcal{O}|$ is a constant, say a, independent of the G-orbit $\mathcal{O}$ in $\mathcal{P}^{\{2\}}$, if and only if each pair of points lies in a constant number of the k-subsets in λ^G. An easy counting argument shows that this number is $a|\lambda^G|$. Thus λ is a base line for a linear space if and only if the constant a is $1/b$ where $b = |\lambda^G|$, that is, if and only if (i) and (ii) hold.

We conclude this discussion with an elementary observation.

Lemma 2.5 *For a line λ and a subset $A \subseteq \lambda$ with $|A| \geq 2$, the setwise stabiliser G_A of A satisfies $G_A \leq G_\lambda$. In particular, every involution fixes a line.*

Proof. Let α, β be distinct points of A. Since λ is the unique line containing $\{\alpha, \beta\}$ it follows that λ is the unique line containing A. Therefore $G_A \leq G_\lambda$. Let now g be an automorphism of order two. Assume $\alpha^g = \beta$. Then $\beta^g = \alpha$ and therefore g stabilizes the set $\{\alpha, \beta\}$. Hence g fixes the line $\lambda(\alpha, \beta)$.

2.3 Intersection numbers

Let $\mathcal{S} = (\mathcal{P}, \mathcal{L})$ be a line-transitive point-imprimitive linear space with respect to a group G of automorphisms and a nontrivial partition $\mathcal{C}$ with d classes of size c, and let $\lambda \in \mathcal{L}$. Then the *intersection numbers* defined as

$$d_i = \Big|\left\{ C \in \mathcal{C} : \ |C \cap \lambda| = i \right\}\Big|, \tag{4}$$

for $0 \leq i \leq k$ are independent of the choice of the line λ and satisfy

$$\sum_{i=1}^{k} d_i \binom{i}{2} = x. \tag{5}$$

The *intersection type* is the vector $(d_0, d_1, \ldots, d_k)$.

3 Constructing linear spaces

Suppose we are given a transitive, imprimitive permutation group G on a set $\mathcal{P}$ of size v and an integer $k > 2$ such that $b = \binom{v}{2}/\binom{k}{2} \in \mathbb{Z}$. We consider here the problem of deciding computationally whether or not there exists a line-transitive linear space $(\mathcal{P}, \mathcal{L})$ for some G-orbit $\mathcal{L}$ on $\mathcal{P}^{\{k\}}$. As noted in Section 1, $\mathcal{L}$ is completely described by one of its members, called a base line. Moreover, Lemma 2.4 contains a criterion, called the orbit condition, for deciding whether $(\mathcal{P}, \lambda^G)$ is a linear space, for a given $\lambda \in \mathcal{P}^{\{k\}}$. We shall describe a procedure for finding, up to an equivalence, all k-subsets λ satisfying the orbit condition.

By the definition of a linear space, any pair of points lies in a unique line, so we may assume that a base line contains a given pair of points. Our strategy is to build up a base line, point by point. If A is an f-subset encountered in this process (where $f \leq k$), A cannot be extended to a k-subset satisfying the orbit condition unless

$$\mu(\mathcal{O}, A) \leq \frac{|\mathcal{O}|}{b} \tag{6}$$

for all G-orbits $\mathcal{O}$ in $\mathcal{P}^{\{2\}}$, where $\mu(\mathcal{O}, A)$ is defined as in (3). This condition is called the *partial orbit condition* and is clearly satisfied by all subsets of base lines. Note however that a k-subset satisfying the partial orbit condition is not necessarily a base line since a base line λ must also satisfy $|G_\lambda| = \frac{|G|}{b}$, where G_λ denotes the setwise stabiliser of λ in G.

3.1 Search Trees and the partial intersection condition

In order to check the orbit condition on all k-subsets of $\mathcal{P}$, a backtrack search may be employed. The first step in the backtrack search is to place a total ordering on $\mathcal{P}$. This order induces a natural lexicographical ordering on subsets of $\mathcal{P}$.

Definition 3.1 Let $(\mathcal{P}, \leq)$ be a finite totally ordered set and consider two subsets $A = \{\alpha_1, \alpha_2, \cdots, \alpha_f\}$ and $B = \{\beta_1, \beta_2, \cdots, \beta_\ell\}$, where

$\alpha_1 < \alpha_2 < \cdots < \alpha_f$ and $\beta_1 < \beta_2 < \cdots < \beta_\ell$. Then $A \le B$ if and only if either

(i) $f \le \ell$ and $\alpha_i = \beta_i$ for all $i \le f$, or

(ii) there exists $i \le \min(f, \ell)$ such that $\alpha_i < \beta_i$, and $\alpha_j = \beta_j$ for all j satisfying $1 \le j < i$.

This search then places all f-subsets of $\mathcal{P}$ with $f \le k$ into a tree such that the children of a node A with $|A| < k$ are all subsets $A \cup \{\alpha\}$ with $\alpha \in \mathcal{P} \setminus A$ and A lexicographically preceding $A \cup \{\alpha\}$.

The reason for this lexicographic condition is the following. The nodes of the tree correspond bijectively to the subsets of $\mathcal{P}$ of size at most k. The root of the tree is the empty set, which is the only set of size 0. In general, for $i \le k$, the sets of size i are at level i of the tree, and the k-subsets of $\mathcal{P}$ form the leaves of this tree. The order in which the nodes are encountered when passing through the tree is exactly the lexicographic order on subsets, provided that we arrange the descendants at any given node in the order which is defined on $\mathcal{P}$. For example, the subsets of size at most 3 of the set $\{1, 2, 3, 4, 5\}$, in lexicographical order, are $\emptyset, \{1\}, \{1, 2\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 5\}, \{1, 3\}, \{1, 3, 4\}, \{1, 3, 5\}, \{1, 4\}, \{1, 4, 5\}, \{1, 5\}, \{2\}, \{2, 3\}, \{2, 3, 4\}, \{2, 3, 5\}, \{2, 4\}, \{2, 4, 5\}, \{3\}, \{3, 4\}, \{3, 4, 5\}$.

If a node fails the partial orbit condition, then none of its descendants will pass it, so the tree may be pruned at that node. If a leaf is found to pass the partial orbit condition and its setwise stabiliser in G is of the required size, then it satisfies the conditions of Lemma 2.4 and thus it is a base line.

Unfortunately the number $\binom{v}{k}$ of k-subsets is in almost all cases too large to perform a backtrack search using only the orbit condition. There are two further conditions that are employed to reduce the search further.

The first of these relates to the Delandtsheer-Doyen Theorem 2.1, while the second is more group theoretic and is discussed in Section 3.2. Suppose that $\mathcal{C}$ is a non-trivial, G-invariant partition of

$\mathcal{P}$ with d classes of size c and that x and y are the corresponding Delandtsheer-Doyen parameters as in Theorem 2.1. It is possible to find all feasible intersection types $(d_0, d_1, \cdots, d_k)$ for a base line, that is, integer solutions to (5). For a given intersection type, define the *partial intersection condition* on an f-subset A of $\mathcal{P}$ by

$$|\{\text{classes } C \in \mathcal{C} : |C \cap A| = i\}| \leq d_i \text{ for all } i = 0, 1, \ldots, k. \quad (7)$$

Since this condition is dependent on the intersection type, several backtrack searches must be performed, one for each of the feasible intersection types. If a subset fails the partial intersection condition, it cannot be extended to a subset satisfying the given intersection type. Since any base line must correspond to one of the given intersection types, this condition may be used to restrict the backtrack search further. If there is more than one G-invariant partition, the intersection types for each partition give extra restrictions. A special instance of this, discussed in [1], is the existence of a grid structure invariant under G.

3.2 Using Symmetry

In this section we confine our comments to consideration of a single G-invariant partition $\mathcal{C}$ of $\mathcal{P}$. The second additional condition on base lines relies on the fact that we only require a classification of line-transitive linear spaces up to isomorphism. An equivalence relation on base lines may be defined such that two base lines are equivalent if the linear spaces they generate are isomorphic. The search should be restricted such that as few base lines as possible per equivalence class are identified. To do this, we use a group H such that $G \leq H \leq N_{\mathrm{Sym}(\mathcal{P})}(G)$, where $N_{\mathrm{Sym}(\mathcal{P})}(G)$ is the normaliser of G in $\mathrm{Sym}(\mathcal{P})$. (Clearly G is contained in $N_{\mathrm{Sym}(\mathcal{P})}(G)$ and it may be a proper subgroup.)

Given $G, \mathcal{C}$ we define $H := N_{\mathrm{Sym}(\mathcal{P})}(G) \cap \mathrm{Stab}(\mathcal{C})$. This is the largest group with the property above that still preserves the G-invariant partition $\mathcal{C}$. Using this group we may define an equivalence relation on subsets of $\mathcal{P}$.

Definition 3.2 Given two subsets A, B of $\mathcal{P}$ we say A and B are *equivalent*, and write $A \sim B$, if $A^H = B^H$, using H as defined above.

Notice that if λ_1 and λ_2 are both k-subsets of $\mathcal{P}$ and $\lambda_1 \sim \lambda_2$ then $(\mathcal{P}, \lambda_1^G)$ is a linear space if and only if $(\mathcal{P}, \lambda_2^G)$ is. This follows because if $\lambda_1 \sim \lambda_2$ then $\lambda_1 = \lambda_2^h$ for some $h \in H$ and h induces an isomorphism from $(\mathcal{P}, \lambda_2^G)$ to $(\mathcal{P}, \lambda_1^G)$.
Thus we only need to test one representative of each H-orbit on $\mathcal{P}^{\{k\}}$. Properties of H may be used to derive restrictions on the k-subsets we need to test. For example, since $G \leq H$ and G is point transitive, H is transitive on $\mathcal{P}$. Hence the representative λ of an H-orbit in $\mathcal{P}^{\{k\}}$ could be chosen to contain a given $\alpha \in \mathcal{P}$.
For $f \leq k$ and $A \in \mathcal{P}^{\{f\}}$, we call A a *partial base line* if A satisfies the partial orbit and partial intersection conditions, (6) and (7). If A is a partial base line and $h \in H$ then A^h satisfies the partial orbit condition, since h permutes the G-orbits $\mathcal{O}$ in $\mathcal{P}^{\{2\}}$ among themselves, and A^h satisfies the partial intersection condition since h leaves $\mathcal{C}$ invariant. Thus H leaves invariant the set of partial base lines of size f, for each $f \leq k$. As far as possible, we should test at stage f of our search only one f-set from each H-orbit on partial base lines in $\mathcal{P}^{\{f\}}$.

3.3 The Search Algorithm

Recall that to organise the search we place an arbitrary total ordering on the point set $\mathcal{P}$. Since in a linear space any pair of points lies in a unique line, we may assume that the two points α_1, α_2 that are least in this ordering lie in our partial base line, that is to say we only need to consider children of the node $\{\alpha_1, \alpha_2\}$ in our search tree.
At this stage, the set of points that need to be considered for addition to the partial base line $\{\alpha_1, \alpha_2\}$ is $Q = \mathcal{P} \setminus \{\alpha_1, \alpha_2\}$. From our previous remarks, we need only test the least point β in each $H_{\{\alpha_1,\alpha_2\}}$-orbit in Q to determine whether it may be added to $\{\alpha_1, \alpha_2\}$. Selecting the least β is in fact arbitrary, it is sufficient to select any point from each orbit to represent that orbit. Suppose we add the least point β of Q

to $\{\alpha_1, \alpha_2\}$. Since in any linear space admitting G, by Lemma 2.5, $G_{\{\alpha_1,\alpha_2\}}$ leaves invariant the line containing $\{\alpha_1, \alpha_2\}$, if β lies in the line then the whole $G_{\{\alpha_1,\alpha_2\}}$-orbit $A(\beta)$ containing β is also contained in the line. Hence we need to test whether $\{\alpha_1, \alpha_2\} \cup A(\beta)$ is a partial base line. If it is not, we remove from Q the whole $H_{\{\alpha_1,\alpha_2\}}$-orbit $Q(\beta)$ containing β and repeat this test for the least point in $Q \setminus Q(\beta)$. On the other hand, if $A := \{\alpha_1, \alpha_2\} \cup A(\beta)$ is a partial base line then we must repeat this testing procedure with $Q(A) = Q \setminus A(\beta)$ in place of Q, and A in place of $\{\alpha_1, \alpha_2\}$.

In the following algorithm the notation H_A is used to denote the setwise stabiliser of A in H. The algorithm is recursive. The general step of the search takes as input a partial base line A of size at least 2, and a subset $Q \subset \mathcal{P} \setminus A$ and proceeds as follows.

Algorithm 1 Search(A,Q)

Global variables*: H and G, transitive permutation groups on* $\mathcal{P}$ *preserving a known partition* $\mathcal{C}$*, and such that* $G \leq H \leq N_{\mathrm{Sym}(\mathcal{P})}(G) \cap \mathrm{Stab}(\mathcal{C})$. *The line size k. An intersection type and the G-orbits on* $\mathcal{P}^{\{2\}}$.

Input*: A, a partial base line with* $2 \leq |A| < k$*; a nonempty subset* $Q \subset \mathcal{P}$*, such that* $Q \cap A = \emptyset$*, of points that need to be tested for addition to A;*

Output*: Certain base lines* λ *such that* $A \subset \lambda \subseteq A \cup Q$ *if such exist. Each* λ *is the lexicographically least element, subject to containing A and being contained in* $A \cup Q$*, of its equivalence class.*

begin

Find G_A*;*

Select the least point $\beta \in Q$*;*

Find the G_A*-orbit,* $A(\beta)$*, containing* β*;*

 # These points must be contained in Q;

if $A(\beta) \not\subseteq Q$

Call Search(A, $Q \setminus A(\beta)$)*;*

else

if $A \cup A(\beta)$ *satisfies the partial orbit and intersection conditions*

if $|A \cup A(\beta)| = k$
Compute $G_{A \cup A(\beta)}$*;*
if $|G_{A \cup A(\beta)}| = \frac{|G|}{b}$
output $A \cup A(\beta)$ *and continue with the algorithm;*
else if $|A \cup A(\beta)| < k$ *and* $Q \setminus A(\beta) \neq \emptyset$
Call Search$(A \cup A(\beta),\, Q \setminus A(\beta))$*;*
 # Now we consider adding a different G_A*-orbit to* A.
Find H_A*;*
Find the H_A*-orbit* $Q(\beta)$ *containing* β*;*
 # Note that $Q(\beta)$ *may contain points not from* Q.
 # We do not need to continue if there are no more points to consider. **if** $Q \setminus Q(\beta) \neq \emptyset$
Call Search$(A,\, Q \setminus Q(\beta))$*;*
end;

If λ is a base line containing A then G_A is the largest known subgroup of G_λ, by Lemma 2.5. As a result, if $\beta \in \lambda \setminus A$ then $A \cup A(\beta) \subseteq \lambda$. Then since we are seeking only base lines $\lambda \subseteq A \cup Q$, we must have $A(\beta)$ contained in Q. After finishing the search for base lines λ such that $A \cup A(\beta) \subseteq \lambda \subseteq A \cup Q$, we are no longer interested in base lines equivalent to ones satisfying $A \cup A(\beta) \subseteq \lambda \subset A \cup Q$. In particular, we are not interested in base lines containing $(A \cup \{\beta\})^h = A \cup \{\beta^h\}$ for any $h \in H_A$. Thus all points of the H_A-orbit $Q(\beta)$ that lie in Q are removed (if $Q(\beta)$ contains points not from Q, then these points were never going to be considered anyway).
As remarked above, if $A(\beta) \nsubseteq Q$ then we remove $A(\beta)$ from Q and recursively call Search$(A, Q \setminus A(\beta))$. Note that we may not at this stage remove the H_A-orbit $Q(\beta)$ containing β since there may be some other G_A-orbit contained in $Q(\beta)$ that lies inside Q and hence must be tested.
The next result shows that the procedure Search terminates, and that each $k-$subset returned is a base line. Following this we prove a second technical proposition that is critical for our purposes. It shows that Search will output all the base lines containing A that are lexicographically least in their equivalence class, subject to the

condition that they contain A.

Proposition 3.3 *The procedure* Search *functions correctly, namely:*

(i) A call to Search*(A,Q) terminates.*

(ii) If λ is returned by the procedure, then λ is a base line invariant under G_A such that $A \subset \lambda \subseteq A \cup Q$.

Proof. *(i)* Every time a recursive call to Search is made, the cardinality of the second parameter, the set Q, decreases strictly. Since the procedure stops when Q is empty, and it is initially finite, the procedure must terminate after a finite number of calls to Search.
(ii) Suppose λ is returned. From the definition of the procedure, this is the case if and only if $|\lambda| = k$ and λ satisfies the orbit conditions. This means that λ satisfies the conditions of Lemma 2.4 and hence λ is a base line. Further, since we obtain λ by adding points from Q to A, we must have $A \subset \lambda \subseteq (A \cup Q)$. To complete the proof we must show that λ is invariant under G_A.
Let A_i denote the first parameter of the ith recursive call of Search in Algorithm 1. We claim that $G_A \leq G_{A_i}$ for all i. From this claim it will follow that λ is G_A-invariant, since $\lambda = A_i$ for some i, thus completing the proof of part *(ii)*. Initially at $i = 1$ we have $A = A_i$ so our claim holds in this case. Suppose inductively that $G_A \leq G_{A_i}$ for some i, and consider A_{i+1}. There are two possibilities for A_{i+1}. Either $A_{i+1} = A_i$ or $A_{i+1} = A_i \cup A(\beta)$ for some $\beta \in Q$. In the first case, $G_A \leq G_{A_{i+1}} = G_{A_i}$ by our inductive hypothesis. In the second case $A(\beta)$ is a G_{A_i}-orbit, A_i is left invariant by G_{A_i}, and so $G_{A_i} \leq G_{A_{i+1}}$. Thus by our assumption, $G_A \leq G_{A_{i+1}}$. It follows by induction that $G_A \leq G_{A_i}$ for all i. □

Proposition 3.4 *Suppose we are given a totally ordered point set $\mathcal{P}$, transitive permutation groups G and H on $\mathcal{P}$, where $G \leq H$ and H normalises G, and sets $A, Q \subset \mathcal{P}$ with $A \cap Q = \emptyset$. Suppose λ is a base line such that λ has the required intersection type for* Search*,*

$A \subset \lambda \subseteq A \cup Q$, and λ is the lexicographically least element of λ^H subject to $A \subset \lambda$. Also let α be the least element of $\lambda \setminus A$. Then either

(i) $\lambda = A \cup (\alpha^{G_A})$, and this set will be returned by a call to SEARCH*(A, Q), or*

(ii) $\lambda \neq A \cup (\alpha^{G_A})$, so $A \cup (\alpha^{G_A}) \subsetneq \lambda$ and a call to SEARCH*(A, Q) will result in a recursive call to* SEARCH*$(A \cup \alpha^{G_A}, Q_1)$ where $\alpha^{G_A} \subseteq Q$ and $\lambda \setminus (A \cup \alpha^{G_A}) \subseteq Q_1 \subseteq Q$.*

In all cases a call to SEARCH*(A,Q) will return λ.*

Proof. We will prove that part (i) or (ii) holds for SEARCH(A, Q) by induction on the number m of points in Q which are less than α. The first case is $m = 0$.
In this case the least point β of Q is α, so $\beta \in \lambda \setminus A$. Since $G_A \leq G_\lambda$ by Lemma 2.5 we must have $A(\beta) = \beta^{G_A} \subseteq (\lambda \setminus A) \subseteq Q$, so the algorithm will consider $A(\beta) \cup A$. Now $A \cup A(\beta) \subseteq \lambda$ and hence $A \cup A(\beta)$ is a partial base line. There are two possibilities for how the algorithm will handle this case. If $A \cup A(\beta) = \lambda$ then, since λ is a base line and hence will pass the stabiliser check performed by SEARCH, λ will be outputted at this stage. Otherwise, since there exists some $\gamma \in \lambda \setminus (A \cup A(\beta)) \subseteq Q \setminus A(\beta)$, the set $Q \setminus A(\beta)$ is non-empty, so SEARCH(A, Q) will call SEARCH$(A \cup \alpha^{G_A}, Q \setminus \alpha^{G_A})$. Clearly $\lambda \setminus (A \cup \alpha^{G_A}) \subseteq (Q \setminus \alpha^{G_A})$ as required. Thus the assertion holds for $m = 0$.
Now suppose the assertion holds for all integers less than a given $m > 0$ and that there are m points in Q less than α. Take β to be the least point in Q. Note that $\beta < \alpha$. We claim that $A(\beta) \cap \lambda = \emptyset$. If there exists $\gamma \in \lambda \cap A(\beta)$ then λ contains $\gamma^{G_A} = \beta^{G_A} = A(\beta)$ so $\beta \in \lambda$, contradicting the minimality of α. There are two possibilities for the behaviour of SEARCH(A, Q).
If $A(\beta) \not\subseteq Q$ the algorithm will call SEARCH$(A, Q \setminus A(\beta))$. Since $\lambda \cap A(\beta) = \emptyset$ we have $\lambda \setminus A \subseteq Q \setminus A(\beta)$. By the inductive hypothesis, since there are fewer than m points less than α in $Q \setminus A(\beta)$, the

assertion holds for $\textsc{Search}(A, Q \setminus A(\beta))$. Since $\textsc{Search}(A, Q)$ calls $\textsc{Search}(A, Q \setminus A(\beta))$, part (i) or (ii) holds true in this case.
Alternatively, if $A(\beta) \subseteq Q$ the algorithm will call $\textsc{Search}(A, Q \setminus Q(\beta))$. We claim $Q(\beta) \cap \lambda = \emptyset$. If $\gamma \in Q(\beta) \cap \lambda$ then $\gamma^h = \beta$ for some $h \in H_A$ and hence $\{\beta\} \cup A \subseteq \lambda^h$. However, since $\beta < \alpha$ and α is the least element of $\lambda \setminus A$ it follows that λ^h is lexicographically less than λ, contradicting the minimality of λ. Thus $Q(\beta) \cap \lambda = \emptyset$ so $\lambda \setminus A \subseteq Q \setminus Q(\beta)$. Notice that there are fewer than m points less than α in $Q \setminus Q(\beta)$, so by the inductive hypothesis, the assertion holds for $\textsc{Search}(A, Q \setminus Q(\beta))$. Since $\textsc{Search}(A, Q)$ calls $\textsc{Search}(A, Q \setminus A(\beta))$, part i) or part ii) holds for $\textsc{Search}(A, Q)$ also.
Thus for all m, part (i) or (ii) holds.
Now we will prove by induction on $\ell = |\lambda \setminus A|$ that $\textsc{Search}(A, Q)$ returns λ. Notice that $\ell \geq 1$, so the first case to prove is $\ell = 1$. Here $\lambda = A \cup \{\alpha\}$ and $\{\alpha\} = \alpha^{G_A} = A(\alpha)$ since λ is G_A-invariant. Thus (i) holds, and a call to $\textsc{Search}(A, Q)$ will output λ.
Suppose that $|\lambda \setminus A| = \ell > 1$ and the claim holds true for all integers less than ℓ. If $\lambda = A \cup \alpha^{G_A}$ then i) holds, so λ is returned by $\textsc{Search}(A, Q)$. Hence we may assume that $\lambda \neq A \cup \alpha^{G_A}$. Thus part (ii) holds, so a call to $\textsc{Search}(A, Q)$ will call $\textsc{Search}(A \cup \alpha^{G_A}, Q_1)$ where $\alpha^{G_A} \subseteq Q$ and $\lambda \setminus (A \cup \alpha^{G_A}) \subseteq Q_1 \subseteq Q$. Notice that $A \cup \alpha^{G_A} \subset \lambda \subseteq Q_1 \cup (A \cup \alpha^{G_A}) \subseteq Q \cup A$, so the conditions of the lemma are also satisfied for this call. However, $|\lambda \setminus (A \cup \alpha^{G_A})| < \ell$, so by the inductive hypothesis, λ is returned by $\textsc{Search}(A \cup \alpha^{G_A}, Q_1)$ and hence by $\textsc{Search}(A, Q)$. □

Algorithm 1 is the crucial part of our procedure to search for the least base line in each equivalence class. In order to use the algorithm to search for all base lines up to equivalence we use the following protocol.

Algorithm 2 Input: *Point set $\mathcal{P}$, permutation group G on $\mathcal{P}$ with a non-trivial G-invariant partition $\mathcal{C}$, line size k, a list of intersection types, a list of G-orbits on pairs of points.*
Output: *The least base line in each equivalence class of base lines.*

begin
Choose a total ordering on $\mathcal{P}$*;*
Find α_1 *and* α_2*, the two smallest points in* $\mathcal{P}$*;*
Calculate $H = N_{\mathrm{Sym}(\mathcal{P})}(G) \cap \mathrm{Stab}(\mathcal{C})$*;*
Set global variables G*,* H *and* k *for* SEARCH*;*
for *each intersection type*
Call SEARCH*(*$\{\alpha_1, \alpha_2\}$*,*$\mathcal{P} \setminus \{\alpha_1, \alpha_2\}$*);*
end for
end;

Theorem 3.5 *Given a point set* $\mathcal{P}$ *and a permutation group* G *preserving a non-trivial partition of* $\mathcal{P}$*, a call to Algorithm 2 will output, up to isomorphism, a base line for each* G*-line transitive,* G*-point imprimitive linear space on* $\mathcal{P}$*.*

Proof. Let $(\mathcal{P}, \mathcal{L})$ be such a linear space. Let α_1, α_2 be the two least elements of $\mathcal{P}$ under the ordering imposed by Algorithm 2. Let $\lambda \in \mathcal{L}$. Then λ is a base line. We will show that Algorithm 2 outputs the lexicographically least element λ_1 of λ^H.
Now $\lambda_1 = \lambda^h$ for some $h \in H$, and $\mathcal{S}_1 := (\mathcal{P}, \lambda_1^G) = (\mathcal{P}, \mathcal{L}^h)$ is a linear space isomorphic to $(\mathcal{P}, \mathcal{L})$ admitting a line-transitive action of $G^h = G$. Therefore some line of $\mathcal{S}_1$, say λ_1^g with $g \in G$, contains the two points α_1, α_2, by the definition of a linear space. As λ_1 is the least element of λ^H it follows that $\lambda_1 \leq \lambda_1^g$ and by the definition of the lexicographic ordering this implies that $\{\alpha_1, \alpha_2\} \subseteq \lambda_1$. However, since $\{\alpha_1, \alpha_2\}$ lies in a unique line of $\mathcal{S}_1$, it follows that $\lambda_1 = \lambda_1^g$. Let $\{d_0, d_1, \cdots, d_k\}$ be the intersection type of λ_1. When this intersection type is considered, Algorithm 2 calls SEARCH$(\{\alpha_1, \alpha_2\}, \mathcal{P} \setminus \{\alpha_1, \alpha_2\})$. Setting, $A = \{\alpha_1, \alpha_2\}$ and $Q = \mathcal{P} \setminus \{\alpha_1, \alpha_2\}$, we have $A \subset \lambda_1 \subseteq A \cup Q$, and hence by Proposition 3.4 this call to SEARCH must output λ_1. □

We conclude with some remarks about possible modifications of this algorithm.

Remark 3.6 (a) Sometimes the calculation of the order of G_λ is not necessary in SEARCH. We know from design theory that $|\lambda| = ab$ for some integer a and hence $|G_\lambda| = \frac{|G|}{ab}$. It is thus sufficient to know that $|G_\lambda| \geq \frac{|G|}{b}$. As we saw in Lemma 3.3, $G_{\{\alpha_1,\alpha_2\}} \leq G_\lambda$. Thus if $|G_{\{\alpha_1,\alpha_2\}}| = \frac{|G|}{b}$, we are guaranteed that G_λ has the correct size.
(b) Generally, α_1 and α_2 are chosen to maximise the size of this stabiliser. In many cases this is large enough that the check is not needed in SEARCH.
(c) The procedure SEARCH calls for up to two setwise stabilisers G_A, H_A to be calculated in each recursive call. Computing set stabilisers can be computationally expensive. To overcome this, we may use known subgroups of these groups. For example, instead of G_A being recalculated at every step, $G_{\{\alpha_1,\alpha_2\}}$ could be used instead. As we saw earlier, $G_{\{\alpha_1,\alpha_2\}} \leq G_A$, so this is a valid substitution, but will result in more recursive calls being required, since less points may be added to the partial base line at each stage. Similarly, it is possible to use the subgroup, $H_{\{\alpha_1,\alpha_2\}}$ of H_A.

References

[1] Anton Betten, Greg Cresp, Alice C. Niemeyer, and Cheryl E. Praeger. Searching for line-transitive linear spaces preserving a grid structure on points. *preprint*, 2004.

[2] Richard E. Block. On the orbits of collineation groups. *Math. Z.*, 96: 33–49, 1967.

[3] Peter J. Cameron and Cheryl E. Praeger. Block-transitive t-designs. I. Point-imprimitive designs. *Discrete Math.*, 118(1-3):33–43, 1993.

[4] C. J. Colbourn and M. J. Colbourn. Cyclic Steiner systems having multiplier automorphisms. *Utilitas. Math.*, 17:127–149, 1980.

[5] Anne Delandtsheer and Jean Doyen. Most block-transitive t-designs are point-primitive. *Geom. Dedicata*, 29(3):307–310, 1989.

[6] Anne Delandtsheer, Alice C. Niemeyer, and Cheryl E. Praeger. Finite line-transitive linear spaces: parameters and normal point partitions. *Adv. Geom.*, 3:469–485, 2003.

[7] W. H. Mills. Two new block designs. *Utilitas Math.*, 7:73–75, 1975.

[8] Werner Nickel and Alice C. Niemeyer. Two families of block-transitive, point-imprimitive block designs. *ANU Mathematical Sciences Research Centre Report*, (SMS-022-90), 1990.

[9] Werner Nickel, Alice C. Niemeyer, Christine M. O'Keefe, Tim Penttila, and Cheryl E. Praeger. The block-transitive, point-imprimitive 2-(729,8,1) designs. *Appl. Algebra Eng. Commun. Comput.*, 3(1):47–61, 1992.

[10] Cheryl E. Praeger and Ngo Dac Tuan. Inequalities involving the Delandtsheer-Doyen parameters for finite line-transitive linear spaces. *Journal of Combinatorial Theory. Series A*, 102:38–62, 2003.

Address of authors:

Anton Betten
Department of Mathematics
Colorado State University
Fort Collins, CO 80523
U.S.A.
Email: betten@math.colostate.edu

Gregory Cresp, Alice C. Niemeyer and Cheryl E. Praeger
School of Mathematics and Statistics
The University of Western Australia
35 Stirling Hwy
Crawley, WA 6009
Australia.
Email: gcresp@maths.uwa.edu.au
alice@maths.uwa.edu.au
praeger@maths.uwa.edu.au