

Sudoku, gerechte designs, resolutions, affine space, spreads, reguli, and Hamming codes

R. A. Bailey, Peter J. Cameron

School of Mathematical Sciences, Queen Mary, University of London,
London E1 4NS, UK

Robert Connelly*

Department of Mathematics, Malott Hall, Cornell University,
Ithaca, NY 14853, USA

Abstract

Solving a Sudoku puzzle involves putting the symbols $1, \dots, 9$ into the cells of a 9×9 grid partitioned into 3×3 subsquares, in such a way that each symbol occurs just once in each row, column, or subsquare. Such a solution is a special case of a gerechte design, in which an $n \times n$ grid is partitioned into n regions with n squares in each, and each of the symbols $1, \dots, n$ occurs once in each row, column, or region. Gerechte designs originated in statistical design of agricultural experiments, where they ensure that treatments are fairly exposed to localised variations in the field containing the experimental plots.

In this paper we consider several related topics. In the first section, we define gerechte designs and some generalizations, and explain a computational technique for finding and classifying them. The second section looks at the statistical background, explaining how such designs are used for designing agricultural experiments, and what additional properties statisticians would like them to have.

In the third section, we focus on a special class of Sudoku solutions which we call “symmetric”. They turn out to be related to some important topics in finite geometry over the 3-element field, and to

*This research partially supported by NSF Grant Number DMS-0510625.

error-correcting codes. We explain all of these connections, and use them to classify the symmetric Sudoku solutions (there are just two, up to the appropriate notion of equivalence). In the final section, we construct some further Sudoku solutions with desirable statistical properties, and briefly consider some generalizations.

1 Gerechte designs

1.1 Introduction

A *Latin square* of order n is an $n \times n$ array containing the symbols $1, \dots, n$ in such a way that each symbol occurs once in each row and once in each column of the array. We say that two Latin squares L_1 and L_2 of order n are *orthogonal* to each other if, given any two symbols i and j , there is a unique pair (k, l) such that the (k, l) entries of L_1 and L_2 are i and j respectively.

In 1956, W. U. Behrens [4] introduced a specialisation of Latin squares which he called “gerechte”. The $n \times n$ grid is partitioned into n regions S_1, \dots, S_n , each containing n cells of the grid; we are required to place the symbols $1, \dots, n$ into the cells of the grid in such a way that each symbol occurs once in each row, once in each column, and once in each region.

The row and column constraints say that the solution is a Latin square, and the last constraint restricts the possible Latin squares.

By this point, many readers will recognize that solutions to Sudoku puzzles are examples of gerechte designs, where $n = 9$ and the regions are the 3×3 subsquares. (The Sudoku puzzle was invented, with the name “number place”, by Harold Garns in 1979.)

Here is another familiar example of a gerechte design. Let L be any Latin square of order n , and let the region S_i be the set of cells containing the symbol i in the square L . A gerechte design for this partition is precisely a Latin square orthogonal to L . (This shows that there is not always a gerechte design for a given partition. A simpler negative example is obtained by taking one region to consist of the first $n - 1$ cells of the first row and the n th cell of the second row.)

We might ask: given a grid, and a partition into regions, what is the complexity of deciding whether a gerechte design exists?

For another example, consider the partitioned grid shown in Figure 1: this example was considered by Behrens in 1956. (Ignore the triples to the

right of the grid for a moment.) Six solutions are shown. Up to rotations of the grid and permutations of the symbols $1, \dots, 5$, these are all the solutions, as we will explain shortly. (The complete set of fifteen solutions is given in [3].)

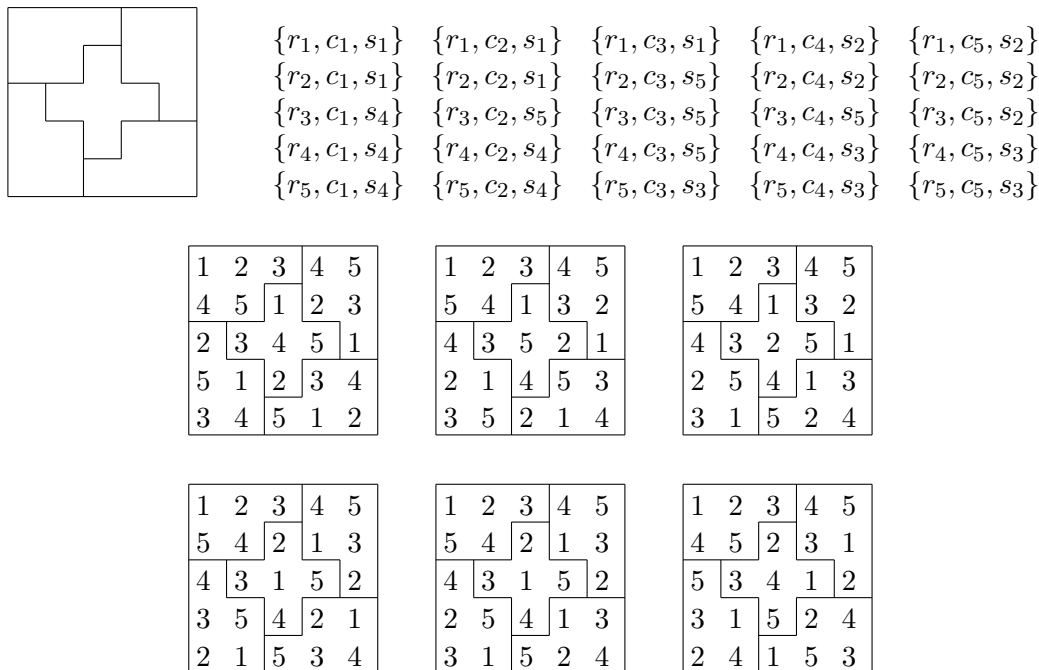


Figure 1: A partitioned 5×5 grid (top left), its representation as a block design (top right), and all inequivalent gerechte designs (bottom)

1.2 Resolvable block designs

A *block design* is a structure consisting of a set of *points* and a set of *blocks*, with an incidence relation between points and blocks. Often we identify a block with the set of points incident to it, so that a block design is represented by a family of sets; however, the same set may occur more than once.

A block design is said to be *resolvable* if the set of blocks can be partitioned into subsets C_1, \dots, C_r (called *replicates*) such that each point is

incident with just one block in any replicate C_i . The partition of the block set is called a *resolution* of the design.

The search for gerechte designs for a given partitioned grid can be transformed into a search for resolutions of a block design, as we now show.

The basic data for a gerechte design is an $n \times n$ grid partitioned into n regions S_1, \dots, S_n , each containing n cells. We can represent this structure by a block design as follows:

- the points are $3n$ objects $r_1, \dots, r_n, c_1, \dots, c_n, s_1, \dots, s_n$;
- for each of the n^2 cells of the grid, there is a block $\{r_i, c_j, s_k\}$, if the cell lies in the i th row, the j th column, and the k th region.

Proposition 1.1 *Gerechte designs on a given partitioned grid correspond, up to permuting the symbols $1, \dots, n$, in one-to-one fashion with resolutions of the above block design.*

Proof Given a gerechte design, let C_i be the set of cells containing the symbol i . By definition, the blocks corresponding to these cells contain each row, column, or region object exactly once, and so form a partition of the point set. Any cell contains a unique symbol i , so every block occurs in just one class C_i . Thus we have a resolution. The converse is proved in the same way. \square

The GAP [10] share package DESIGN [20] can find all resolutions of a block design, up to isomorphisms of the block design. In our case, isomorphisms of the block design come from symmetries of the partitioned grid, so we can use this package to compute all gerechte designs up to permutation of symbols and symmetries of the partitioned grid.

For example, the partition of the 5×5 grid discussed in the preceding section is represented as a block design with 15 points and 25 blocks of size 3, also shown in Figure 1. The automorphism group of the design is the cyclic group of order 4 consisting of the rotations of the grid through multiples of $\pi/2$. The DESIGN program quickly finds that, up to automorphisms, there are just six resolutions of this design, corresponding to six inequivalent gerechte designs; these are shown in the figure.

The same method shows that, for a 6×6 square divided into 3×2 rectangles, there are 49 solutions up to symmetries of the corresponding block design and permutations of the symbols. (The number of symmetries of the block design in this case is 3456; the group consists of all row and column permutations preserving the appropriate partitions.)

1.3 Orthogonal and multiple gerechte designs

We saw earlier the definition of orthogonality of Latin squares. A set of *mutually orthogonal* Latin squares is a set of Latin squares in which every pair is orthogonal. It is known that the size of a set of mutually orthogonal Latin squares of order n is at most $n - 1$.

Similar definitions and results apply to gerechte designs. We say that two gerechte designs with the same partitioned grid are *orthogonal* to each other if they are orthogonal as Latin squares, and a set of *mutually orthogonal* gerechte designs is a set of such designs in which each pair is orthogonal.

Proposition 1.2 *Given a partition of the $n \times n$ grid into regions S_1, \dots, S_n each of size n , the size of a set of mutually orthogonal gerechte designs for this partition is at most $n - d$, where d is the maximum size of the intersection of a region S_i and a line (row or column) $L_j \neq S_i$.*

Proof Take a cell $c \in L_j \setminus S_i$. By permuting the symbols in each square, we may assume that all the squares have entry 1 in the cell c . Now, in each square, the symbol 1 occurs exactly once in the region S_i and not in the line L_j ; and all these occurrences must be in different cells, since for each pair of squares, the pair (1, 1) of entries already occurs in cell c . So there are at most $|S_i \setminus L_j|$ squares in the set. \square

This bound is not always attained. Consider the 5×5 gerechte designs given earlier. The maximum intersection size of a line and a region is clearly 3, so the bound for the number of mutually orthogonal designs is 2. But by inspection, each design has the property that the entries in cells (2, 3) and (3, 5) are equal. (The reader is invited to discover the simple argument to show that this must be so, independent of the classification of the designs.) Hence no pair of orthogonal designs is possible. Similarly, for the 6×6 square divided into 3×2 rectangles, there cannot exist two orthogonal gerechte designs, since it is well known that there cannot exist two orthogonal Latin squares of order 6.

Proposition 1.2 gives an upper bound of 6 for the number of mutually orthogonal Sudoku solutions. In Section 3.4, we will see that this bound is attained.

The concept of a gerechte design can be generalized. Suppose that we are given a set of r partitions of the cells of an $n \times n$ grid into n regions each of

size n . A *multiple gerechte design* for this partition is a Latin square which is simultaneously a gerechte design for all of the partitions.

For example, given a set of (mutually orthogonal) Latin squares, the symbols in each square define a partition of the $n \times n$ array into regions. A Latin square is a multiple gerechte design for all of these partitions if and only if it is orthogonal to all the given Latin squares.

The problem of finding a multiple gerechte design can be cast into the form of finding a resolution of a block design, in the same way as for a single gerechte design. The block design has $(r + 2)n$ points, and each cell of the grid is represented by a block containing the objects indexing its row, its column, and the region of each partition which contain it. Again, we can use the DESIGN program to classify such designs up to symmetries of the grid.

For example, Federer [9], in a section which he attributed to G. M. Cox, called a $m_1 m_2 \times m_1 m_2$ Latin square *magic* if it is a gerechte design for the regions forming the obvious partition into $m_1 \times m_2$ rectangles, and *super magic* if it is simultaneously a gerechte design for the partition into $m_2 \times m_1$ rectangles, where $m_1 \neq m_2$. He considered the problem of finding multiple gerechte designs (which he called “super magic Latin squares”) for the 6×6 square partitioned into 3×2 rectangles and 2×3 rectangles. The DESIGN package finds that there are 26 such designs up to symmetries.

We can also define a set of mutually orthogonal multiple gerechte designs in the obvious way, and prove a similar bound for the size of such a set.

We will see examples of these things in Section 3.4.

2 Statistical considerations

In this section, we consider the use of gerechte designs in statistical design theory, and some additional properties which are important there.

2.1 Agricultural experiments in Latin squares

The statistician R. A. Fisher suggested the use of Latin squares in agricultural experiments. If n “treatments” (crop varieties, quantities of fertilizer, etc.) are to be compared on plots forming an $n \times n$ grid in a field, then arranging the treatments as the symbols of a Latin square ensures that any systematic change in fertility, drainage, etc. across the field affects all treatments equally. Figure 2 shows two experiments laid out as Latin squares.



Figure 2: Two experiments using Latin squares. Left: a 5×5 forestry experiment in Beddgelert in Wales, to compare varieties of tree; designed by Fisher, laid out in 1929, and photographed in about 1945. Right: a current 6×6 experiment to compare methods of controlling aphids; conducted by Lesley Smart at Rothamsted Research, photographed in 2004.

If a Latin square experiment is to be conducted on land that has recently been used for another Latin square experiment, it is sensible to regard the previous treatments as relevant and so to use a Latin square orthogonal to the previous one. As explained above, this is technically a sort of gerechte design, but no agricultural statistician would call it that.

The purpose of a gerechte design in agricultural experimentation is to ensure that all treatments are fairly exposed to any different conditions in the field. In fact, “gerecht(e)” is the German for “fair” in the sense of “just”. Rows and columns are good for capturing differences such as distance from a wood but not for marking out stony patches or other features that tend to clump in compact areas. Thus, in the statistical and agronomic literature, the regions of a gerechte design are always taken to be “spatially compact” areas.

2.2 Randomization

Before a design is used for an experiment, it is *randomized*. This means that a permutation of the cells is chosen at random from among all those that preserve the three partitions: into rows, into columns, and into regions. It is by no means common for the cells to be actually square plots on the ground; when they are, it is also possible to transpose rows and columns, if the regions are unchanged by this action. This random permutation is applied to the chosen gerechte design before it is laid out in the field.

One important statistical principle is *lack of bias*. This means that every plot in the field should be equally likely to be matched, by the randomization, to each abstract cell in the gerechte design, so that any individual plot with strange characteristics is equally likely to affect any of the treatments. To achieve this lack of bias, the set of permutations used for randomizing must form a *transitive* group, in the sense that there is such a permutation carrying any nominated cell to any other. The allowable permutations of the 5×5 grid in Figure 1 do not have this property, but those for magic Latin squares do. There are others, but no complete classification as far as we know.

For the remainder of this section we assume that $n = m_1 m_2$ and the regions are $m_1 \times m_2$ rectangles. Then the rows, columns and regions define some other areas: a *large row* is the smallest area that is simultaneously a union of regions and a union of rows; a *minirow* is the non-empty intersection of a row and region; *large columns* and *minicolumns* are defined similarly.

A pair of distinct cells in such a grid is in one of eight relationships, illustrated in Figure 3 for the 6×6 grid with 3×2 regions. For $i = 1, \dots, 8$, the cell labelled $*$ is in relationship i with the cell labelled i . Thus a pair of distinct cells is in relationship 1 if they are in the same minirow; relationship 2 if they are in the same minicolumn; relationship 3 if they are in the same region but in different rows and columns; relationship 4 if they are in the same row but in different regions; relationship 5 if they are in the same column but in different regions; relationship 6 if they are in the same large row but in different rows and regions; relationship 7 if they are in the same large column but in different columns and regions; relationship 8 if they are in different large rows and large columns.

The group of permutations used for randomization has the property that a pair of distinct cells can be mapped to another pair by one of the permutations if and only if they are in the same relationship. If, in addition, we can transpose the rows and columns (not possible in Figure 3) then relationships

*	1		4		
2				6	
	3				
5					8
	7				

Figure 3: Eight relationships between pairs of distinct cells in the 6×6 grid

1 and 2 are merged, as are 4 and 5, and 6 and 7.

The simple-minded analysis of data from an experiment in a gerechte design assumes that the response (such as yield of grain, or the logarithm of the number of aphids) on each cell is the sum of four unknown parameters, one each for the row, column and region containing the cell, and one for the treatment (symbol) applied to it. In addition, there is random variation from cell to cell. This is explained in [2]. The statistician is interested in the treatment parameters, not only in their values but also in whether their differences are greater than can be explained by cell-to-cell variation.

However, one school of statistical thought holds that if the innate differences between rows, between columns and between regions are relevant, then so potentially are those between minirows, minicolumns, large rows and large columns. Yates took this view in his 1939 paper [24], whose discussion of a 4×4 Latin square “with balanced corners” may be the first published reference to gerechte designs. Thus the eight relationships all have to be considered when the gerechte design is chosen.

2.3 Orthogonality and the design key

Two further important statistical properties often conflict with each other. One is ease of analysis, which means not ease of performing arithmetic but ease of explaining the results to a non-statistician. So-called *orthogonal* designs, like the one in Figure 4, have this property.

A gerechte design with rectangular regions is orthogonal if the arrangement of symbols in each region can be obtained from the arrangement in any other region just by permuting minirows and minicolumns. In Figure 4,

5	2	6	3	4	1
6	3	4	1	5	2
4	1	5	2	6	3
2	5	3	6	1	4
3	6	1	4	2	5
1	4	2	5	3	6

Figure 4: An orthogonal design for the 6×6 grid with 3×2 regions

each minicolumn contains either treatments 1, 2 and 3 or treatments 4, 5 and 6. When the statistician investigates whether there is any real difference between the average effects of these two sets of treatments, (s)he compares their difference (estimated from the data) with the underlying variability between minicolumns within regions and columns (also estimated from the data). Similarly, differences between the average effects of the three sets of two treatments $\{1, 4\}$, $\{2, 5\}$ and $\{3, 6\}$ are compared with the variability of minirows within regions and rows. Treatment differences orthogonal to all of those, such as the difference between the average of $\{1, 5\}$ and the average of $\{2, 4\}$, are compared with the residual variability between the cells after allowing for the variability of all the partitions.

An orthogonal design for an $m_1 m_2 \times m_1 m_2$ square with $m_1 \times m_2$ regions may be constructed using the *design key* method [21, 22], as recommended in [3]. The large rows are labelled by A_1 , which takes values $1, \dots, m_2$. Within each large row, the rows are labelled by A_2 , which takes values $1, \dots, m_1$. Similarly, the large columns are labelled by B_1 , taking values $1, \dots, m_1$, and the columns within each large column by B_2 , taking values $1, \dots, m_2$. Then put $N_1 = A_1 + B_2$ modulo m_2 and $N_2 = A_2 + B_1$ modulo m_1 . The ordered pairs of values of N_1 and N_2 give the $m_1 m_2$ symbols. In Figure 4, the rows are numbered from top to bottom, the columns from left to right, and the correspondence between the ordered pairs and the symbols is as follows.

		N_2		
N_1		1	2	3
1		1	2	3
2		4	5	6

(When explaining this construction to non-mathematicians we usually take the integers modulo m to be $1, \dots, m$ rather than $0, \dots, m - 1$.)

Variations on this construction are possible, especially when m_1 and m_2 are both powers of the same prime p . For example, if $m_1 = 4$ and $m_2 = 2$ then we can work modulo 2, using A_1 to label the large rows, A_2 and A_3 to label the rows within large rows, B_1 and B_2 to label the large columns, and B_3 to label the columns within large columns. Numbers can be allocated by putting $N_1 = A_1 + B_3$, $N_2 = A_2 + B_1$ and $N_3 = A_3 + B_2$. All that is required is that no non-zero linear combination (modulo 2) of N_1 , N_2 and N_3 contains only A_1 , B_1 and B_2 , or a subset thereof.

2.4 Efficiency and concurrence

The other important statistical property is *efficiency*, which means that the estimators of the differences between treatments should have small variance. At one extreme, we might decide that the innate differences between mini-columns are so great that the design in Figure 4 provides no information at all about the difference between the average of treatments 1, 2, 3 and the average of treatments 4, 5, 6; and similarly for minirows. In this case, it can be shown (see [1, Chapter 7]) that the relevant variances can be deduced from the matrix

$$M = m_1 m_2 I - \frac{1}{m_2} \Lambda_R - \frac{1}{m_1} \Lambda_C + J.$$

Here I is the $n \times n$ identity matrix and J is the $n \times n$ all-1 matrix. The *concurrence* of symbols i and j in minirows is the number of minirows containing both i and j (which is n when $i = j$): the matrix Λ_R contains these concurrences. The matrix Λ_C is defined similarly, using concurrences in mini-columns. It is known that if the off-diagonal entries in the matrix M are all equal then the average variance is as small as possible for the given values of m_1 and m_2 , so the usual heuristic is to choose a design in which the off-diagonal entries differ as little as possible. If $m_1 = m_2$, this means that the sums of the concurrences are as equal as possible. We explore this property for Sudoku solutions in Section 4.1.

A compromise between these two statistical properties is *general balance* [13, 16, 17], which requires that the concurrence matrices Λ_R and Λ_C commute with each other. A special case of general balance is *adjusted orthogonality* [8, 14], for which $\Lambda_R \Lambda_C = n^2 J$. It can be shown that a gerechte design

with rectangular regions is orthogonal in the sense of Section 2.3 if it has adjusted orthogonality and $\Lambda_R^2 = nm_2\Lambda_R$ and $\Lambda_C^2 = nm_1\Lambda_C$. This property is also explored further in Section 4.1.

3 Some special Sudoku solutions

Our main aim in this section is to consider some very special Sudoku solutions which we call *symmetric*. We state our main results first. The proofs will take us on a tour through parts of finite geometry and coding theory; we have included brief introductions to these topics, for readers unfamiliar with them who want to follow us through the proofs of the theorems.

We have seen that a Sudoku solution is a gerechte design for the 9×9 array partitioned into nine 3×3 subsquares. To define symmetric Sudoku solutions, we need a few more types of region.

As defined in the last section, a *minirow* consists of three cells forming a row of a subsquare, and a *minicolumn* consists of three cells forming a column of a subsquare. We define a *broken row* to be the union of three minirows occurring in the same position in three subsquares in a column, and a *broken column* to be the union of three minicolumns occurring in the same position in three subsquares in a row. A *location* is a set of nine cells occurring in a fixed position in all of the subsquares (for example, the centre cells of each subsquare).

Now a *symmetric Sudoku solution* is an arrangement of the symbols $1, \dots, 9$ in a 9×9 grid in such a way that each symbol occurs once in each row, column, subsquare, broken row, broken column, and location. In other words, it is a multiple gerechte design for the partitions into subsquares, broken rows, broken columns, and locations. Figure 5 shows a symmetric Sudoku solution. The square shown has the further property that each of the 3×3 subsquares is “semi-magic”, that is, its row and column sums (but not necessarily its diagonal sums) are 15 (John Bray [6]).

As in the first section, two Sudoku solutions are *equivalent* if one can be obtained from the other by a combination of row and column permutations (and possibly transposition) which preserve all the relevant partitions, and re-numbering of the symbols.

The main result of this section asserts that, up to equivalence, there are precisely two symmetric Sudoku solutions. This theorem can be proved by a computation of the type described in the first section. However, we give a

8	1	6	2	4	9	5	7	3
3	5	7	6	8	1	9	2	4
4	9	2	7	3	5	1	6	8
7	3	5	1	6	8	4	9	2
2	4	9	5	7	3	8	1	6
6	8	1	9	2	4	3	5	7
9	2	4	3	5	7	6	8	1
1	6	8	4	9	2	7	3	5
5	7	3	8	1	6	2	4	9

Figure 5: A semi-magic symmetric Sudoku solution

more conceptual proof, exploiting the links with the other topics of the title.

We also consider mutually orthogonal sets; we show that the maximum number of mutually orthogonal Sudoku solutions is 6, and the maximum number of mutually orthogonal symmetric Sudoku solutions is 4. Moreover, there is a set of six mutually orthogonal Sudoku solutions of which four are symmetric. These are exhibited in Figure 10.

Throughout this section will use $\text{GF}(3)$ to denote the finite field with three elements (the integers modulo 3).

3.1 Preliminaries

In this subsection we describe briefly the notions of affine and projective geometry and coding theory. Readers familiar with this material may skip this subsection.

Affine geometry An *affine space* is just a vector space with the distinguished role of the origin removed. Its subspaces are the *cosets* of the vector subspaces, that is, sets of the form $U + v$, where U is a vector subspace and v a fixed vector, the *coset representative*. This coset is also called the *translate* of U by v . Two affine subspaces which are cosets of the same vector subspace are said to be *parallel*, and the set of all cosets of a given vector subspace forms a *parallel class*. A *transversal* for a parallel class of affine subspaces is a set of coset representatives for the vector subspace.

We use the terms “point”, “line” and “plane” for affine subspaces of dimension 0, 1, 2 respectively. We denote the n -dimensional affine space over a field F by $\text{AG}(n, F)$; if $|F| = q$, we write $\text{AG}(n, q)$.

We will use the fact that a subset of $\text{AG}(n, F)$ is an affine subspace if (and only if) it contains the unique affine line through each pair of its points. In affine space over the field $\text{GF}(3)$, a line has just three points, and the third point on the line through p_1 and p_2 is the “midpoint” $(p_1 + p_2)/2 = -(p_1 + p_2)$.

Projective geometry Much of the argument in the proof of the main theorem of this section will be an examination of collections of subspaces of a vector space. This can also be cast into geometric language, that of projective geometry.

The n -dimensional *projective space* over a field F is the geometry whose points, lines, planes, etc. are the 1-, 2-, 3-dimensional (and so on) subspaces of an $(n + 1)$ -dimensional space V (which we can take to be F^{n+1}). A point P lies on a line L if $P \subset L$ (as subspaces of F^{n+1}).

For example, a point of the projective space $\text{PG}(n, F)$ is a 1-dimensional subspace of the vector space F^{n+1} , and so it corresponds to a parallel class of lines in the affine space $\text{AG}(n + 1, F)$. The points of the projective space can therefore be thought of as “points at infinity” of the affine space.

We will mostly be concerned with 3-dimensional projective geometry; we refer to [7, 12]. We will use the following notions:

- Two lines are said to be *skew* if they are not coplanar. Skew lines are necessarily disjoint. Conversely, since any two lines in a projective plane intersect, disjoint lines are skew. So the terms “disjoint” and “skew” for lines in projective space are synonyms. We will normally refer to disjoint lines. Note that disjoint lines in $\text{PG}(n, F)$ arise from 2-dimensional subspaces in F^{n+1} meeting only in the origin.
- A hyperbolic quadric is a set of points satisfying an equation like $x_1x_2 + x_3x_4 = 0$. Any such quadric contains two “rulings”, each of which is a set of pairwise disjoint lines covering all the points of the quadric (Figure 6). Such a set of lines is called a *regulus*, and the other set is the *opposite regulus*. Any three pairwise disjoint lines of the projective space lie in a unique regulus. The lines of the opposite regulus are all the lines meeting the given three lines (their *common transversals*).

- A *spread* is a family of pairwise disjoint lines covering all the points of the projective space. A spread is *regular* if it contains the regulus through any three of its lines. (Any three lines of a spread are pairwise disjoint, and so lie in a unique regulus.) It can be shown that, if the field F is finite, then there exists a regular spread. In particular, this holds when $F = \text{GF}(3)$.

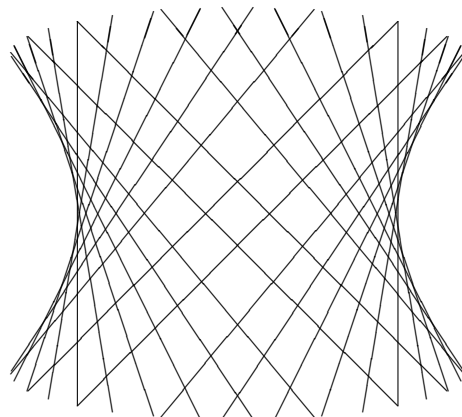


Figure 6: A hyperbolic quadric and its two rulings

The fact that any pair of lines in a projective plane intersect is a consequence of the dimension formula of linear algebra. The points and lines of the plane are 1- and 2-dimensional subspaces of a 3-dimensional vector space; and if two 2-dimensional subspaces U_1 and U_2 are unequal, then

$$\dim(U_1 \cap U_2) = \dim(U_1) + \dim(U_2) - \dim(U_1 + U_2) = 2 + 2 - 3 = 1.$$

The second and third bullet points are most easily proved using coordinates. We will see an example of a regulus and its opposite in coordinates later. In cases where regular spreads exist, any three pairwise disjoint lines are contained in a regular spread.

In the final section of the paper we briefly consider higher dimensions, and use the fact that $\text{PG}(2m - 1, F)$ has a spread of $(m - 1)$ -dimensional subspaces.

Coding theory A *code* of length n over a fixed alphabet A is just a set of n -tuples of elements of A ; its members are called *codewords*. The *Hamming distance* between two n -tuples is the number of positions in which they

differ. The *minimum distance* of a code is the smallest Hamming distance between distinct codewords. For example, if the minimum distance of a code is 3, and the code is used in a communication channel where one symbol in each codeword might be transmitted incorrectly, then the received word is closer to the transmitted word than to any other codeword (by the triangle inequality), and so the error can be corrected; we say that such a code is 1-error-correcting.

A 1-error-correcting code of length 4 over an alphabet of size 3 contains at most 9 codewords. For, given any codeword, there are $1 + 4 \cdot 2 = 9$ words which can be obtained from it by making at most one error; these sets of nine words must be pairwise disjoint, and there are $3^4 = 81$ words altogether, so there are at most 9 such sets. If the bound is attained, the code is called *perfect*, and has the property that any word is distant at most 1 from a unique codeword.

It is known that there is, up to a suitable notion of equivalence, a unique perfect code of length 4 over an alphabet of size 3, the so-called *Hamming code*. We do not assume this uniqueness; we will determine all perfect codes in the course of our proof (see Proposition 3.2).

If the alphabet is a finite field F , the code C is *linear* if it is a subspace of the vector space F^n . The Hamming code is a linear code. Note that translation by a fixed vector preserves Hamming distance; so, for example, if a linear code is perfect 1-error-correcting, then so is each of its cosets.

A linear code C of dimension k can be specified by a *generator matrix*, a $k \times n$ matrix whose row space is C . The code with generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix} \tag{1}$$

is a Hamming code. Of course, permutations of the rows and columns of this matrix, and multiplication of any column by -1 , give generator matrices for other Hamming codes.

See Hill [11] for further details.

3.2 Sudoku and geometry over $\text{GF}(3)$

Following the idea of the design key described in Section 2.3, we coordinatize the cells of a Sudoku grid using $\text{GF}(3) = \{0, 1, 2\}$. Each cell c has four coordinates (x_1, x_2, x_3, x_4) , where

- x_1 is the number of the large row containing c ;
- x_2 is the number of the minirow of this subsquare which contains c ;
- x_3 is the number of the large column containing c ;
- x_4 is the number of the minicolumn of this subsquare which contains c .

(In each case we start the numbering at zero. We number rows from top to bottom and columns from left to right.)

Now the cells are identified with the points of the four-dimensional vector space $V = \text{GF}(3)^4$. The origin of the vector space is the top left cell. However, there is nothing special about this cell, so we should think of the coordinates as forming an affine space $\text{AG}(4, 3)$.

Some regions of the Sudoku grid which we have already discussed are cosets of 2-dimensional subspaces, as shown in the following table. Each 2-dimensional subspace corresponds to a line in $\text{PG}(3, 3)$; we name these lines for later reference.

Equation	Description of cosets	Line in $\text{PG}(3, 3)$
$x_1 = x_2 = 0$	Rows	L_1
$x_3 = x_4 = 0$	Columns	L_2
$x_1 = x_3 = 0$	Subsquares	L_3
$x_1 = x_4 = 0$	Broken columns	L_5
$x_2 = x_3 = 0$	Broken rows	L_6
$x_2 = x_4 = 0$	Locations	L_4

Table 1: Some subspaces of $\text{GF}(3)^4$

In addition, the main diagonal is the subspace defined by the equations $x_1 = x_3$ and $x_2 = x_4$, and the antidiagonal is $x_1 + x_3 = x_2 + x_4 = 2$, a coset of the subspace $x_1 = -x_3, x_2 = -x_4$. (The other cosets of these two subspaces are not so obvious in the grid.)

Now, in a Sudoku solution, each symbol occurs in nine positions forming a transversal for the cosets of the subspaces defining rows, columns, and subsquares as above (this condition translates into “one position in each row, column, or subsquare”). A Sudoku solution is symmetric if it also has the analogous property for broken rows, broken columns, and locations.

We call a Sudoku solution *linear* if, for each symbol, its nine positions form an affine subspace in the affine space. All the Sudoku solutions in this subsection and the next are linear. We will say that a linear Sudoku solution is of *type A* if all nine affine subspaces are cosets of the same vector subspace, and of *type B* otherwise.

3.3 Symmetric Sudoku solutions

In this section we classify, up to equivalence, the symmetric Sudoku solutions. We show that there are just two of them; both are linear, and one is of type A (defined by the nine cosets of a fixed subspace), while the other is of type B (involving cosets of different subspaces).

Consider the set of positions where a given symbol occurs in a symmetric Sudoku solution, regarded as a subset of $V = \text{GF}(3)^4$. These positions form a code of length 4 containing nine codewords. Given any two coordinates i and j , and any two field elements a and b , there is a unique codeword p satisfying $p_i = a$ and $p_j = b$ (see Table 1). The minimum distance of this code is thus at least 3, since distinct codewords cannot agree in two positions. Conversely, if S is a set of points with minimum distance at least 3, then for any given a and b , there is at most one $p \in S$ with $p_i = a$ and $p_j = b$; so, if $|S| = 9$, there must be exactly one such point p . So we have shown:

Proposition 3.1 *A symmetric Sudoku solution is equivalent to a partition of V into nine perfect codes.*

It is clear from this Proposition that the partition into cosets of a Hamming code gives a symmetric Sudoku solution. We prove that there is just one further partition, up to equivalence.

Proposition 3.2 *Any perfect 1-error correcting code in $V = \text{GF}(3)^4$ is an affine subspace.*

Proof Let H be such a perfect code. Then H consists of 9 vectors, any two agreeing in at most one coordinate. As above, given distinct coordinates i, j and field elements a, b , there is a unique $p \in H$ with $p_i = a$ and $p_j = b$.

Any two vectors of H have distance at least 3; so

$$\sum_{p, q \in H} d(p, q) \geq 9 \cdot 8 \cdot 3 = 216,$$

where d denotes Hamming distance. On the other hand, if we choose any coordinate position (say the first), and suppose that the number of vectors of H having entries 0, 1, 2 there are respectively n_0, n_1, n_2 , then the contribution of this coordinate to the above sum is

$$n_0(9 - n_0) + n_1(9 - n_1) + n_2(9 - n_2) = 81 - (n_0^2 + n_1^2 + n_2^2) \leq 81 - 27 = 54,$$

and so the entire sum is at most $4 \cdot 54 = 216$. So equality must hold, from which we conclude that any pair of vectors have distance 3 (that is, agree in one position).

Now take $p, q \in H$. Suppose, without loss, that they agree in the first coordinate; say $p = (a, b_1, c_1, d_1)$ and $q = (a, b_2, c_2, d_2)$. Since $b_1 \neq b_2$, the remaining element of $\text{GF}(3)$ is $-(b_1 + b_2)$. There is a unique element r of H having first coordinate a and second coordinate $-(b_1 + b_2)$; since it must disagree with each of p and q in the third and fourth coordinates, it must be $r = (a, -(b_1 + b_2), -(c_1 + c_2), -(d_1 + d_2)) = -(p + q)$. This is the third point on the affine line through p and q . So H is indeed an affine subspace, as required (cf. p. 14). \square

Any translate of a perfect code is a perfect code; so any perfect code is a coset of a vector subspace which is itself a perfect code. We call such a subspace *allowable*. Our next task is to find the allowable subspaces.

Lemma 3.3 *The vectors $p = (a_1, a_2, a_3, a_4)$, and $q = (b_1, b_2, b_3, b_4)$ are two linearly independent vectors in an allowable subspace X of V if and only if the four ratios a_i/b_i , for $i = 1, 2, 3, 4$ are distinct, where $\pm 1/0 = \infty$ is one ratio that must appear, and the indeterminate form $0/0$ does not appear.*

Proof The vectors p, q and any two of the standard basis vectors, with just one non-zero coordinate (equal to 1), must be linearly independent. So the determinant of the corresponding matrix, which is $a_i b_j - a_j b_i$, is not zero. Then the result follows. \square

Given Lemma 3.3, we see that when a basis for an allowable subspace is put into row-reduced echelon form, it takes one the following eight possibilities.

$$\left\{ \left\{ \begin{array}{c} 1011 \\ \text{or} \\ 1022 \end{array} \right\} \text{ and } \left\{ \begin{array}{c} 0112 \\ \text{or} \\ 0121 \end{array} \right\} \right\} \text{ or } \left\{ \left\{ \begin{array}{c} 1012 \\ \text{or} \\ 1021 \end{array} \right\} \text{ and } \left\{ \begin{array}{c} 0111 \\ \text{or} \\ 0122 \end{array} \right\} \right\} \quad (2)$$

These are the only allowable subspaces. So any perfect code in V is a coset of one of those eight vector subspaces.

Our conclusion for symmetric Sudoku solutions so far can be summarized as follows:

- Any symmetric Sudoku solution is linear;
- In a symmetric Sudoku solution, the positions of each symbol form a coset of one of the eight allowable subspaces.

Next we come to the question of how such subsets can partition V . One simple way is just to take all cosets of one of the above 2-dimensional vector subspaces; this gives the solutions we described above as Type A. Another choice is the following. Extend an allowable subspace X to an appropriate 3-dimensional vector subspace Y of V . The three cosets of Y partition V , and we can look for another allowable subspace X' of Y which can be used to partition one or two of these cosets. For this to work, it is necessary that the linear span of X and X' be 3-dimensional. For each choice of an allowable X , it is easy to check that there are four other allowable X' such that the span of X and X' is 3-dimensional, but there is no set of three allowable subspaces such that the span of each pair is 3-dimensional.

Conversely, take any symmetric Sudoku solution, and consider the corresponding partition of V into cosets of allowable 2-dimensional subspaces. If any pair of such subspaces are distinct and span the whole of V , then any of their cosets will intersect, contradicting the Sudoku property. Thus their span must be a 3-dimensional vector subspace Y and hence they are two subspaces X and X' as in the previous paragraph. Thus X and X' are the only two allowable subspaces parallel to any set in the partition for a Sudoku solution. Furthermore, in each of the three cosets of Y , cosets of only one of X or X' can appear. Thus the Sudoku solutions described in the previous paragraph are the only ones possible.

Using this analysis we can see that for each choice of one of the 8 allowable planes, since there are exactly 4 choices for another such that their span is 3-dimensional, there are $8 \cdot 4/2 = 16$ possible choices of such pairs. For each pair, we want to use each plane to partition at least one of the three 3-dimensional affine spaces determined by the pair of planes: there are 6 ways of doing this. Thus there are $6 \cdot 16 = 96$ possible Sudoku solutions of this sort. In addition, there are 8 Type A Sudoku solutions comprising the cosets of a single plane. This gives $96 + 8 = 104$ total number of symmetric Sudoku

solutions, falling into just two classes up to equivalence under symmetries of the grid.

In the spirit of the Sudoku puzzle, we give in Figure 7 a partial symmetric Sudoku which can be uniquely completed (in such a way that each row, column, subsquare, broken row, broken column or location contains each symbol exactly once). The solution is of type B; that is, it is not equivalent to the one shown in Figure 5.

							7
			7				
	6						
	4		3				
		1	5				8
				2		7	
				1	4		
				4			
1							

Figure 7: A Sudoku-type puzzle

The fact that there are just two inequivalent symmetric Sudoku solutions, proved in the above analysis, can be confirmed with the DESIGN program, which also shows that, if we omit the condition on locations, there are 12 different solutions; and, if we omit both locations and broken columns, there are 31021 different solutions. The total number of Sudoku solutions up to equivalence (that is, solutions with only the conditions on rows, columns, and subsquares) is 5472730538; this number was computed by Ed Russell and Frazer Jarvis [18].

3.4 Mutually orthogonal Sudoku solutions

In this section we construct sets of mutually orthogonal Sudoku solutions of maximum size. The results of the construction are shown in Figure 10.

Theorem 3.4 (a) *There is a set of six mutually orthogonal Sudoku solutions. These squares are also gerechte designs for the partition into*

locations, and have the property that each symbol occurs once on the main diagonal and once on the antidiagonal. Each of the Sudoku solutions is linear of type A.

- (b) There is a set of four mutually orthogonal multiple gerechte designs for the partitions into subsquares, locations, broken rows and broken columns; they also have the property that each symbol occurs once on the main diagonal and once on the antidiagonal. Each of the Sudoku solutions is linear of type A.

Remark We saw already that the number 6 in part (a) is optimal. The number 4 in (b) is also optimal. For, given such a set, we can as before suppose that they all have the symbol 1 in the cell in the top left corner. Now the 1s in the subsquare in the middle of the top row cannot be in its top minirow or its left-hand minicolumn, so just four positions are available; and the squares must have their ones in different positions.

Proof (a) Our six Sudoku solutions will all be linear of type A; that is, they will be given by six parallel classes of planes in the affine space. The orthogonality of two solutions means that each plane of the first meets each plane of the second in a single point. This holds precisely when the two vector subspaces meet just in the origin (so that their direct sum is the whole space). In other words, the vector subspaces correspond to disjoint lines in the projective space $\text{PG}(3, 3)$.

In our situation, the affine planes $x_1 = x_2 = 0$ and $x_3 = x_4 = 0$ whose cosets define rows and columns correspond to two disjoint lines L_1 and L_2 of $\text{PG}(3, 3)$; and the affine plane $x_1 = x_3 = 0$ whose cosets define the subsquares to a line L_3 which intersects both L_1 and L_2 (in the points $\langle(0, 0, 0, 1)\rangle$ and $\langle(0, 1, 0, 0)\rangle$ respectively). So we have to find six pairwise disjoint lines which are disjoint from the given three lines.

Now there is a regulus \mathcal{R} containing L_1 and L_2 , whose opposite regulus contains L_3 . Moreover, \mathcal{R} is contained in a regular spread. Then the six lines of the spread not in \mathcal{R} are disjoint from L_3 , and have the required property. (See Figure 8.)

Calculation shows that the remaining lines of \mathcal{R} are $x_1 - x_3 = x_2 - x_4 = 0$ and $x_1 + x_3 = x_2 + x_4 = 0$, and the other three lines of the opposite regulus are $x_1 - x_2 = x_3 - x_4 = 0$, $x_1 + x_2 = x_3 + x_4 = 0$, and $x_2 = x_4 = 0$, which is the Locations line L_4 (the line such that the cosets of the corresponding vector

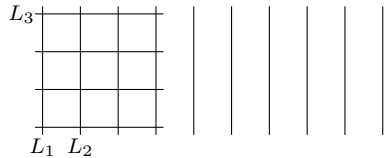


Figure 8: A regulus, the opposite regulus, and a spread

subspace define the partition into locations). The main diagonal and the antidiagonal are cosets of the subspaces corresponding to the other two lines of \mathcal{R} . Since the remaining six lines of the spread are disjoint from these, our claim about locations and diagonals follows. It is clear from the construction that all the corresponding Sudoku solutions are linear of type A.

A different set of six mutually orthogonal Sudoku solutions can be obtained by choosing a regulus \mathcal{R}^* disjoint from \mathcal{R} and contained in the spread, and replacing it by the opposite regulus. This also gives linear solutions of type A.

(b) For the second part, it is more convenient to work in the affine space $\text{AG}(4, 3)$. As we have seen, a type A symmetric Sudoku solution is given by the cosets of one of the eight allowable subspaces of V . It is easily checked that the following four matrices span subspaces with the property that any two of them meet only in the zero vector, from which it follows that the corresponding symmetric Sudoku solutions are orthogonal.

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 2 & 2 \\ 1 & 0 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 2 & 2 \end{bmatrix}$$

Another set of four mutually orthogonal symmetric Sudoku solutions is obtained by using the other four allowable subspaces (obtained by changing the sign of the coordinates in the final column). \square

We can use the solution to (b) to find an explicit construction for (a). Recall that we seek six lines of the projective space disjoint from the lines L_1, L_2 and L_3 . All of these must be disjoint from L_4 also.

Four of these are also disjoint from the lines L_5 and L_6 defined by $x_1 = x_4 = 0$ and $x_2 = x_3 = 0$; these are the four Hamming codes H_1, \dots, H_4 that we constructed. Now, there is a unique regulus \mathcal{R}' containing L_1 and

L_2 and having L_5 and L_6 in the opposite regulus; the other two lines of \mathcal{R}' can be added to the four lines arising from the Hamming codes to produce the required set of six lines. They have equations $x_1 + x_4 = x_2 + x_3 = 0$ and $x_1 - x_4 = x_2 - x_3 = 0$. See Figure 9. The resulting six mutually orthogonal Sudoku solutions are shown in Figure 10; the last four are symmetric.

It can be shown that the four lines H_1, \dots, H_4 disjoint from the two reguli themselves form a regulus; they and the lines of the opposite regulus are the eight Hamming codes.

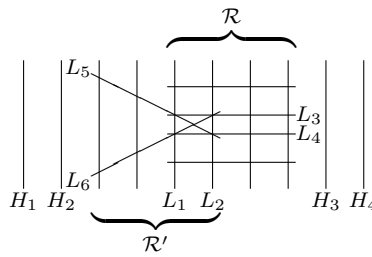


Figure 9: Two reguli in the construction of mutually orthogonal gerechte designs

111	222	333	749	857	968	475	586	694
111	222	333	658	469	547	896	974	785
444	555	666	173	281	392	718	829	937
444	555	666	982	793	871	239	317	128
777	888	999	416	524	635	142	253	361
777	888	999	325	136	214	563	641	452
326	134	215	952	763	841	687	498	579
589	697	478	734	815	926	342	153	261
659	467	548	385	196	274	921	732	813
823	931	712	167	248	359	675	486	594
983	791	872	628	439	517	354	165	246
256	364	145	491	572	683	918	729	837
238	319	127	864	945	756	593	671	482
965	746	854	273	381	192	427	538	619
562	643	451	297	378	189	836	914	725
398	179	287	516	624	435	751	862	943
895	976	784	531	612	423	269	347	158
632	413	521	849	957	768	184	295	376

Figure 10: Six mutually orthogonal Sudoku solutions

This analysis can also be used to define and count orthogonal symmetric Sudoku solutions. First we note that, if two symmetric Sudoku solutions are orthogonal, then both must be of type A. For, as we saw earlier, orthogonality means that each coset in the first solution meets each coset in the second in a single affine point (so the corresponding lines in the projective space are disjoint). A type B Sudoku solution involves cosets of two 2-dimensional spaces with non-zero intersection, corresponding to two intersecting lines in $\text{PG}(3, 3)$. But the only lines available are the eight lines of a regulus and its opposite, and no such line is disjoint from two intersecting lines in the set.

Now two type A symmetric Sudoku solutions are orthogonal if and only if the corresponding lines belong to the same regulus. So there are $8 \cdot 3 = 24$ such ordered pairs.

4 Further special Sudoku solutions and generalizations

In the first subsection of this section, we construct some Sudoku solutions having some of the desirable statistical properties defined in Section 2. In the second, we give some generalizations to gerechte designs of other sizes, using other finite fields.

4.1 The block design in minirows and minicolumns

The cells in the minirows and minicolumns form lines of the affine space $\text{AG}(4, 3)$. In any type A symmetric Sudoku solution comprising all cosets of a fixed vector subspace S , such a line together with S spans a 3-dimensional subspace which contains three cosets of S . So all the nine lines in this subspace contain the same three symbols. This means that the 27 minirows define just three triples from $\{1, \dots, 9\}$, each triple occurring in nine minirows. The same condition holds for the minicolumns. Thus the design is orthogonal, in the sense of Section 2.3. Moreover, the block design on $\{1, \dots, 9\}$ formed by the minirows and minicolumns is a 3×3 grid with each grid line occurring nine times as a block. Each pair of symbols lies in either 0 or 9 blocks of the design. (These properties are easily verified by inspection of Figure 5.)

In general, a block design is said to be *balanced* if every pair of symbols lies in the same number of blocks. Since the average number of blocks containing a pair of symbols from $\{1, \dots, 9\}$ in this design is $2 \cdot 27 \cdot 3 / \binom{9}{2} = 9/2$, the design

cannot be balanced. But we could ask whether there is a Sudoku solution which is better balanced than a type A symmetric solution; for example, one in which each pair occurs in either 4 or 5 blocks. Such solutions exist; the first example was constructed by Emil Vaughan [23].

Given such a design with pairwise concurrences 4 and 5, we obtain a regular graph of valency 4 on the vertex set $\{1, \dots, 9\}$ by joining two vertices if they occur in five blocks of the design. The “nicest” such graph is the 3×3 grid, the line graph of $K_{3,3}$. (This graph is strongly regular, and the resulting design would be partially balanced with respect to the Hamming association scheme consisting of the graph and its complement: see [1].) Vaughan’s solution does not realize this graph, but we subsequently found one which does. An example is given in Figure 11. (Two vertices in the same row or column of the 3×3 grid are adjacent.)

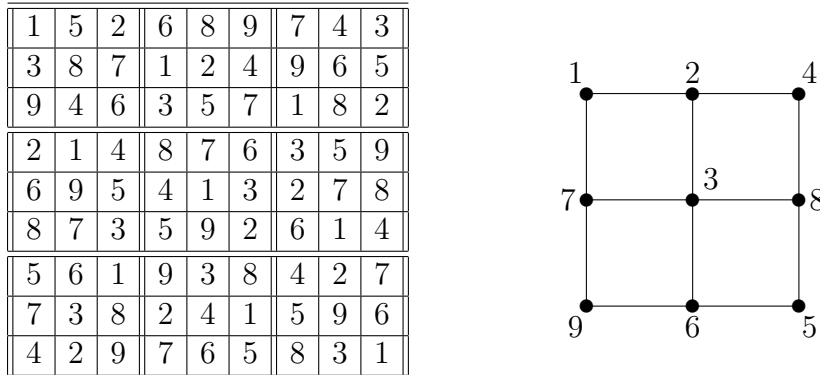


Figure 11: A Sudoku solution in which the block design in minirows and minicolumns has concurrences 4 and 5, and its corresponding graph

We could ask whether even more is true: is there a Sudoku solution in which each pair of symbols occur together 2 or 3 times in a minirow, 2 or 3 times in a minicolumn, and 4 or 5 times altogether? (We saw in Section 2.4 that balancing concurrences in minirows and minicolumns separately is a desirable statistical property.) A computation using **GAP** showed that such a solution cannot exist; one cannot place more than five symbols satisfying these constraints without getting stuck. It is not clear what the “best” compromise is.

We further found that there exist Sudoku solutions in which the design in minirows and minicolumns is partially balanced with respect to the 3×3 grid with concurrences $(4, 5)$, $(3, 6)$, $(2, 7)$ or $(0, 9)$, but not $(1, 8)$ (for which at most four symbols can be placed). The type A linear Sudoku solution in Figure 5 realizes the case $(0, 9)$.

We also considered another special type of Sudoku solution based on the properties of the minirows and minicolumns: those for which the designs formed by minirows and minicolumns have adjusted orthogonality, in the sense that their concurrence matrices Λ_R and Λ_C satisfy $\Lambda_R \Lambda_C = 81J$, where J is the all-one matrix. (Here the (i, j) entry of Λ_R counts the number of minirows in which i and j both occur, and similarly for Λ_C .) The special Sudoku solution of Figure 5 has this property, but it is not unique. (In this solution, all entries of each concurrence matrix are 0 or 9.) We found that there are, up to symmetry, 194 Sudoku solutions for which the minirows and minicolumns have adjusted orthogonality in this sense, of which 104 have the property that both Λ_R and Λ_C have entries different from 0 and 9. One of these solutions is shown in Figure 12.

1	2	3	4	5	6	7	8	9
7	8	9	1	3	2	6	5	4
4	5	6	7	8	9	1	3	2
3	1	2	6	4	5	9	7	8
9	7	8	2	1	3	4	6	5
6	4	5	9	7	8	2	1	3
8	9	1	5	6	4	3	2	7
2	3	7	8	9	1	5	4	6
5	6	4	3	2	7	8	9	1

Figure 12: Minirows and minicolumns form designs with adjusted orthogonality, but the overall design is not orthogonal

A word about the computations reported in this section. The strategy is to place the symbols $1, \dots, 9$ in the grid successively to satisfy the constraints. The positions of a single symbol in the grid subject to the Sudoku constraints that it occurs once in each row, column and subsquare can be described by a permutation π of the set $\{1, \dots, 9\}$, where the set of positions

is $\{(i, \pi(i)) : 1 \leq i \leq 9\}$. There are 6^6 of these ‘‘Sudoku permutations’’. We say that two Sudoku permutations are ‘‘compatible’’ if they place their symbols in disjoint cells satisfying the appropriate conditions (for example, for concurrences 4 and 5, that there are either 4 or 5 occurrences of the two symbols in the same minirow or minicolumn). Then we form a graph as follows: the vertex set is the set of all Sudoku permutations, and we join two vertices if they are compatible. We now search randomly for a clique of size 9 in the compatibility graph: this is a set of nine mutually compatible Sudoku permutations, defining a Sudoku solution with the required properties.

Adjusted orthogonality of the two designs is not captured by any obvious compatibility condition on the Sudoku permutations, and we proceeded differently. Since each of the two concurrence matrices has diagonal entries 9, we see that adjusted orthogonality implies that two symbols cannot occur both in the same minirow and in the same minicolumn. Using this as the compatibility condition, we built the compatibility graph, and found all cliques of size 9, using the GAP package GRAPE [19]. Remarkably, it turned out that all of them actually give designs with adjusted orthogonality; we know no simple reason for this fact, since our compatibility condition appears not strong enough to guarantee this.

4.2 Other finite field constructions

The construction in Section 3.4 can be generalized.

Proposition 4.1 *Let q be a prime power, and a and b positive integers. Let $n = q^{a+b}$. Partition the $n \times n$ square into $q^a \times q^b$ rectangles. Then we can find*

$$q^{a+b} - 1 - \frac{(q^a - 1)(q^b - 1)}{q - 1}.$$

mutually orthogonal gerechte designs for this partitioned grid.

Remark If $a < b$, our upper bound for the number of mutually orthogonal gerechte designs for this grid is $q^b(q^a - 1)$. If $a = 1$, this bound is equal to the number in the theorem, so our bound is attained. If $a > 1$, however, the bound is not met by the construction. For example, if $p = 2$, $a = 2$ and $b = 3$, the bound is 24 but the construction achieves 10. If a and b are not coprime, we can improve the construction by replacing q, a, b by $q^d, a/d, b/d$, where $d = \gcd(a, b)$.

Proof Represent the cells by points of the affine space $\text{AG}(2(a+b), q)$ with coordinates $x_1, \dots, x_{a+b}, y_1, \dots, y_{a+b}$. The rows are cosets of the subspace $x_1 = \dots = x_{a+b} = 0$, the columns are cosets of the subspace $y_1 = \dots = y_{a+b} = 0$, and the rectangles are cosets of $x_1 = \dots = x_a = y_1 = \dots = y_b = 0$.

As before, we work in the projective space $\text{PG}(2(a+b) - 1, q)$. The first two subspaces are disjoint, and are part of a spread of $q^{a+b} - 1$ subspaces of the same dimension. The third subspace meets the first in $(q^b - 1)/(q - 1)$ points and the second in $(q^a - 1)/(q - 1)$ points, and has $(q^a - 1)(q^b - 1)/(q - 1)$ further points. In the worst case, this subspace meets $(q^a - 1)(q^b - 1)/(q - 1)$ further spaces of the spread, each in one point. This leaves $q^{a+b} - 1 - (q^a - 1)(q^b - 1)/(q - 1)$ spread spaces disjoint from it, as required. \square

Our construction of mutually orthogonal symmetric Sudoku solutions also generalizes:

Proposition 4.2 *Let q be a prime power, and consider the $q^2 \times q^2$ grid, partitioned into $q \times q$ subsquares, broken rows, broken columns, and locations as in the preceding section. Then there exist $(q - 1)^2$ mutually orthogonal multiple gerechte design for these partitions; this is best possible.*

Proof We follow the same method as before, working over $\text{GF}(q)$. The lines of $\text{PG}(3, q)$ defining rows, columns, subsquares, broken rows, broken columns, and locations lie in the union of two reguli with two common lines, which form part of a regular spread. The remaining $(q - 1)^2$ lines of the spread give the required designs. The upper bound is proved as before. \square

Acknowledgements

The left-hand photograph in Figure 2 appears in [5], “reproduced by permission of the Forestry Commission”. It can also be found on the web at [15]. We thank Lesley Smart for permission to use the right-hand photograph, which was taken by Neil Mason of the Plant and Invertebrate Ecology Division of Rothamsted Research.

We are grateful to the referees for substantial improvements to the presentation of the paper.

References

- [1] R. A. Bailey, *Association Schemes: Designed Experiments, Algebra and Combinatorics*, Cambridge Studies in Advanced Mathematics **84**, Cambridge University Press, Cambridge, 2004.
- [2] R. A. Bailey, J. Kunert and R. J. Martin, Some comments on gerechte designs. I. Analysis for uncorrelated errors. *J. Agronomy & Crop Science* **165** (1990), 121–130.
- [3] R. A. Bailey, J. Kunert and R. J. Martin, Some comments on gerechte designs. II. Randomization analysis, and other methods that allow for inter-plot dependence, *J. Agronomy & Crop Science* **166** (1991), 101–111.
- [4] W. U. Behrens, Feldversuchsanordnungen mit verbessertem Ausgleich der Bodenunterschiede, *Zeitschrift für Landwirtschaftliches Versuchs- und Untersuchungswesen* **2** (1956), 176–193.
- [5] J. F. Box, *R. A. Fisher: The Life of a Scientist*. John Wiley & Sons, New York, 1978.
- [6] J. N. Bray, personal communication, February 2006.
- [7] P. J. Cameron, *Projective and Polar Spaces*, QMW Maths Notes **13**, Queen Mary and Westfield College, London, 1991; available from <http://www.maths.qmul.ac.uk/~pjc/pps/>
- [8] J. A. Eccleston and K. G. Russell, Connectedness and orthogonality in multi-factor designs, *Biometrika* **62** (1975), 341–345.
- [9] W. T. Federer, *Experimental Design—Theory and Applications*, Macmillan, New York, 1955.
- [10] The GAP Group, **GAP** — Groups, Algorithms, and Programming, Version 4.6; Aachen, St Andrews, 2005, <http://www.gap-system.org/>
- [11] R. Hill, *A First Course in Coding Theory*, Clarendon Press, Oxford, 1986.
- [12] J. W. P. Hirschfeld, *Finite Projective Spaces of Three Dimensions*, Oxford University Press, Oxford, 1985.

- [13] A. M. Houtman and T. P. Speed, Balance in designed experiments with orthogonal block structure, *Ann. Statist.* **11** (1983) 1069–1085.
- [14] S. M. Lewis and A. M. Dean, On general balance in row–column designs, *Biometrika* **78** (1991), 595–600.
- [15] Materials for the History of Statistics.
URL: <http://www.york.ac.uk/depts/math/histstat/>
- [16] J. A. Nelder, The analysis of randomized experiments with orthogonal block structure. II. Treatment structure and the general analysis of variance, *Proc. Roy. Soc. London A* **283** (1965) 163–178.
- [17] J. A. Nelder, The combination of information in generally balanced designs, *J. Roy. Statistic. Soc. B* **30** (1968), 303–311.
- [18] Ed Russell and Frazer Jarvis, There are 5472730538 essentially different Sudoku grids,
<http://www.afjarvis.staff.shef.ac.uk/sudoku/sudgroup.html>
- [19] L. H. Soicher, **GRAPE**: a system for computing with graphs and groups. In L. Finkelstein and W. M. Kantor, editors, *Groups and Computation*, volume 11 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 287–291. American Mathematical Society, 1993. **GRAPE** homepage:
<http://www.maths.qmul.ac.uk/~leonard/grape/>
- [20] Leonard H. Soicher, The **DESIGN** package for **GAP**,
http://designtheory.org/software/gap_design/
- [21] H. D. Patterson, Generation of factorial designs, *J. Roy. Statist. Soc. B* **38** (1976), 175–179.
- [22] H. D. Patterson and R. A. Bailey, Design keys for factorial experiments, *Applied Statistics* **27** (1978), 335–343.
- [23] Emil Vaughan, personal communication, November 2005.
- [24] F. Yates, The comparative advantages of systematic and randomized arrangements in the design of agricultural and biological experiments, *Biometrika* **30** (1939), 440–466.