Slide 1

Designs on the Web

Peter J Cameron

p.j.cameron@qmul.ac.uk

Joint work with R. A. Bailey, P. Dobcsányi, J. P. Morgan and L. H. Soicher

http://designtheory.org

Slide 2

Designs on the Web: aims

• Encourage communication between practitioners of combinatorial and statistical design theory; make up-to-date research in combinatorics easily available to statisticians and *vice versa*.

• Provide a standard interchange format (independent of particular software) for designs and their properties: the "External Representation".

• Provide a database of designs usable by both practising statisticians and combinatorial researchers.

• Provide a repository of software and documentation and a discussion forum for all aspects of design theory.

Slide 3

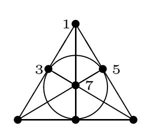
What is a design?

We have a set T of treatments which we wish to compare, and a set Ω of plots or experimental units to which the treatments may be applied.
A *design* is a function $F : \Omega \to T$ (so that $F(\omega)$ is the treatment applied to plot ω).
If T and Ω are completely unstructured, simply use each treatment the same number of times (as near as possible).
The existence of structure on T and Ω complicates matters!

Slide 4

A design

Suppose that we have to test seven different types of fertiliser. Three plots on each of seven farms are available. We use each fertiliser on three plots; the experimental design is the choice of which three plots should receive the same fertiliser.
Clearly it would not be a good idea to use one fertiliser on all three plots of one farm; then the effect on the yield caused by the fertiliser could not be distinguished from the effect caused by farm. (The effects are *confounded*.)
The best design is in fact the following one, which mathematicians recognise as the *projective plane of order* 2:



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

Slide 5

Treatment and plot structures

For example, one treatment may be a placebo, or the “standard” treatment against which the others are to be compared; or the treatments may have a number of factors which can be applied at different levels (fertiliser dose, watering, planting time, etc.) These are important but for now we disregard them. If T is homogeneous, we can replace the function F by a partition of Ω (and assign one treatment to each part).
Plot structure arises because the set of plots is not homogeneous. For example, we may have several plots on each of a number of farms in different geographical areas; soil fertility or moisture content may vary across a field; one plant may shade another; experiments over a period of time are subject to daily or seasonal effects; and drugs may have carry-over effects.

Slide 6

Block designs

The simplest plot structure is a system of blocks, where plots in a block are more alike than those in different blocks (as in the farms example). Thus, the blocks form a partition of Ω. So a *block design* consists of a set Ω with two partitions, corresponding to treatments and blocks.
The block partition is given, and we have to choose the treatment partition so that information can be recovered about the treatments as efficiently as possible – this means that the variances of the estimators of treatment differences should be as small as possible.
In the worst case, if we applied each treatment on only one farm, we couldn’t separate treatment effects from geographical effects.

Slide 7

Representing a design

$\mathbf{R}_1$: A set of plots with two partitions (as above).
$\mathbf{R}_2$: Bipartite graph, vertices are points (treatments) and blocks, joined if the treatment appears in the block, labelled with the number of times it appears.
$\mathbf{R}_3$: Set of points, each block is a multiset of points, so the blocks form a multiset of multisets.
$\mathbf{R}_3'$: Dual to $\mathbf{R}_3$.
$\mathbf{R}_4$: Bipartite graph with vertices and edges labelled. Edge labels as above; vertex labels denote the number of repetitions of blocks and/or points.
The design is called *binary* if no treatment occurs more than once in a block (so edge multiplicities are all 1).

Slide 8

Example

Suppose we have six plots, $1, 2, 3, 4, 5, 6$, which fall into two blocks $\alpha = \{1, 2, 3, 4\}$ and $\beta = \{5, 6\}$. We have two treatments A and B, and apply A to plots $1, 3, 5$ and B to plots $2, 4, 6$. These two partitions form representation $\mathbf{R}_1$. It has 8 automorphisms.
$\mathbf{R}_2$: Complete bipartite graph on vertices $\{A, B\}$ and $\{\alpha, \beta\}$, with multiplicies 2 on $A\alpha$ and $A\beta$ and 1 on the other two edges. This has 2 automorphisms.
$\mathbf{R}_3$: Point set $\{A, B\}$, blocks $[A, A, B, B]$ and $[A, B]$. Again 2 automorphisms.
$\mathbf{R}_3'$: Point set $\{\alpha, \beta\}$; block $\{\alpha, \alpha, \beta\}$ with multiplicity 2. Trivial automorphism group.
$\mathbf{R}_4$: Complete bipartite graph with vertex α labelled 2, and A and B labelled 1; edges $A\alpha$ labelled 2, $B\alpha$ labelled 1. Trivial group.

Slide 9

The structure

$$
\begin{tikzcd}
 & & \mathbf{R}_3 \arrow[dr] & \\
\mathbf{R}_1 \arrow[r] & \mathbf{R}_2 \arrow[ur] \arrow[dr] & & \mathbf{R}_4 \\
 & & \mathbf{R}_3' \arrow[ur] &
\end{tikzcd}
$$

The arrows correspond to homomorphisms between the automorphism groups. The first homomorphism measures the non-binarity of the design; the others measure repeated blocks or points.

For the DTRS project, at present we consider only binary designs. We have chosen to use representation $\mathbf{R}_3$, which is almost universal among mathematicians. Thus, a block design for us consists of a set of points, and a list of blocks, each block a subset of the point set; but repeated blocks are allowed.

Slide 10

Combinatorial and statistical properties

Combinatorialists like the property of *balance*: any two points are contained in a constant number λ of blocks. They consider various strengthenings (such as the t-design condition: any t points are contained in a constant number of blocks) or weakenings such as "partial balance".

Statisticians, on the other hand, care about being able to extract as much information as possible from the experimental data. This depends on properties of the *information matrix*

$$X_T^\top (I - X_b K^{-1} X_B^\top) X_T,$$

where X_T and X_B are the plot-treatment and plot-block incidence matrices. The eigenvalues of this matrix, apart from a "trivial" zero, measure the efficiency of obtaining treatment contrasts, and should be as large as possible.

Slide 11

Robustness

Another feature of a design with real-world relevance is its *robustness*: does it remain reasonably good if data from a few plots, or even a few blocks, is lost? At least we would like to have results asserting bounds on the value of some parameter when a few blocks are lost from the design.

Robustness is related to design construction. Suppose we want all designs which are optimal with respect to some criterion. Rather than list all designs, calculate the relevant parameter for each, and choose the best, we'd like to do a backtrack search, and need to be able to prune the tree. In other words, we need results that say, if I have chosen some of the blocks and calculated some parameter, I can put an upper bound on the value of the parameter however the remaining blocks are chosen. This is "dual" to robustness, and maybe similar techniques can be developed.

Slide 12

External Representation of designs

The External Representation is in XML format. We believe that this is a standardised and well-established format which should be increasingly recognised in the foreseeable future.

A design in XML format is human-readable to a limited extent, but is primarily intended for exchange between programs, databases, etc. The document has a syntax which can be checked; it also makes some mathematical assertions about the designs it contains, which can also in principle be checked.

The External Representation document contains a specification of the syntax together with extensive documentation.

Slide 13

An example

This is a list of designs containing a single design, the Fano plane, in XML.

```xml
<list_of_designs
 design_type="block_design" no_designs="1"
 pairwise_nonisomorphic="true"
 xmlns="http://designtheory.org/xml/ns">
    <block_design b="7" id="ls-t2v7k3lambda1-1" v="7">
        <blocks ordered="true">
            <block><n>0</n><n>1</n><n>2</n></block>
            <block><n>0</n><n>3</n><n>4</n></block>
            <block><n>0</n><n>5</n><n>6</n></block>
            <block><n>1</n><n>3</n><n>5</n></block>
            <block><n>1</n><n>4</n><n>6</n></block>
            <block><n>2</n><n>3</n><n>6</n></block>
            <block><n>2</n><n>4</n><n>5</n></block>
        </blocks>
    </block_design>
</list_of_designs>
```

At the first level of indentation, between the opening and closing tags `block_design`, we have indeed specified the design: it has $v = 7$ points, $b = 7$ blocks, and the seven blocks are listed (the first one is $[0, 1, 2]$, and there are tags to identify each of 0, 1, 2 as natural numbers). There is also a somewhat meaningless identification string for the design.

Slide 14

Specifying a block design

The specification of a block design is shown below. Properties followed by a question mark are optional. Remember that all designs are assumed to be *binary*, so that a block is a set of points (rather than a multiset). This is an example of RNC, a compact and friendly way to specify the syntax of an XML document. The previous example included only the required fields.

```
block_design = element block_design {
    attribute id { xsd:ID } ,
    attribute v  { xsd:positiveInteger } ,
    attribute b  { xsd:positiveInteger } ,
    blocks ,
    point_labels ? ,
    indicators ? ,
    combinatorial_properties ? ,
    automorphism_group ? ,
    resolutions ? ,
    partial_balance_properties ? ,
    statistical_properties ? ,
    alternative_representations ? ,
    info ?
}
```

Slide 15

Designs on the Web: future

At present a version of the external representation exists (in XML with documentation), and we have computed all designs of various sizes and the corresponding optimality parameters.
Future goals include:
• Building the database (first it is necessary to decide on its format).
• Questions about numerical representation of algebraic data such as eigenvalues.
• Developing software to construct and analyse designs including their statistical properties (interfacing with standard algebraic and statistical computing packages via the external representation).
• Questions about types of partial balance of more general designs (for concurrence, variance, and efficiency).

Slide 16

Partial balance

An association scheme is a set of zero-one matrices, containing the identity matrix I and summing to the all-one matrix J, such that each matrix is symmetric and the span of the matrices is closed under multiplication. A block design is *partially balanced* with respect to an association scheme on the set of treatments if the number of blocks containing two treatments depends only on the associate class containing that pair.
If we replace closure under multiplication by closure under *Jordan product* $A \circ B = \frac{1}{2}(AB+BA)$, we obtain the weaker notion of a *Jordan scheme*.
There are also variance and efficiency forms of balance, which have not been extended satisfactorily to partial balance.

Slide 17

Jordan schemes

Association schemes and, later, Jordan schemes, were introduced to simplify the calculation of eigenvalues and eigenspaces of large information matrices. Association schemes are now well-studied: Jordan schemes much less so!
In particular, we do not have any example of a Jordan scheme which is not obtained from a coherent configuration by symmetrisation. (A coherent configuration is like an association scheme but with the requirement of symmetry weakened to the condition that, if A is in the set, so is $A^\top$; its symmetrisation is obtained by replacing each pair $\{A, A^\top\}$ of non-symmetric matrices by $A + A^\top$.)