Relations among partitions.
III: Some structures with three or four partitions

R. A. Bailey
University of St Andrews

Combinatorics Seminar, Shanghai Jiao Tong University,
November 2017
If we insist that all the pairwise relations among the partitions are either orthogonality or balance (in one or both directions) or adjusted orthogonality with respect to a third partition, then we obtain interesting structures such as Youden squares, double Youden rectangles and triple arrays.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

$F \prec G$ means that $F$ is a refinement of $G$, in the sense that every part of $F$ is contained in a single part of $G$ but $F \neq G$. 
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \prec G$ means that $F$ is a refinement of $G$, in the sense that every part of $F$ is contained in a single part of $G$ but $F \neq G$.
- $F \perp G$ means that $F$ is strictly orthogonal to $G$, in the sense that
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \prec G$ means that $F$ is a refinement of $G$, in the sense that every part of $F$ is contained in a single part of $G$ but $F \neq G$.

- $F \perp G$ means that $F$ is strictly orthogonal to $G$, in the sense that
  
  (i) every part of $F$ meets every part of $G$ (so that $F \vee G = U$) and
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \prec G$ means that $F$ is a refinement of $G$, in the sense that every part of $F$ is contained in a single part of $G$ but $F \neq G$.

- $F \perp G$ means that $F$ is strictly orthogonal to $G$, in the sense that
  1. every part of $F$ meets every part of $G$ (so that $F \vee G = \mathcal{U}$)
  and
  2. for each $\omega$ in $\Omega$,

$$\frac{|F(\omega) \cap G(\omega)|}{|\Omega|} = \frac{|F(\omega)|}{|\Omega|} \times \frac{|G(\omega)|}{|\Omega|}.$$
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \prec G$ means that $F$ is a refinement of $G$, in the sense that every part of $F$ is contained in a single part of $G$ but $F \neq G$.

- $F \perp G$ means that $F$ is strictly orthogonal to $G$, in the sense that
  (i) every part of $F$ meets every part of $G$ (so that $F \vee G = U$) and
  (ii) for each $\omega$ in $\Omega$,

\[
\frac{|F(\omega) \cap G(\omega)|}{|\Omega|} = \frac{|F(\omega)|}{|\Omega|} \times \frac{|G(\omega)|}{|\Omega|}.
\]

- $F \perp G$ means that $F$ is orthogonal to $G$, which means that, although $F \vee G$ may not be $U$, the above equation is true with $\Omega$ replaced by $F \vee G(\omega)$. 
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \triangleright G$ means that $F$ is balanced with respect to $G$, in the sense that $N_{FG}N_{GF}$ is completely symmetric with non-zero off-diagonal elements, but $F$ is not strictly orthogonal to $G$. 
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \blacktriangleright G$ means that $F$ is balanced with respect to $G$, in the sense that $N_{FG}N_{GF}$ is completely symmetric with non-zero off-diagonal elements, but $F$ is not strictly orthogonal to $G$.

- $F \blacktriangleright G$ means that $F \blacktriangleright G$ and the relationship between $F$ and $G$ is binary or generalized binary, in the sense that the size of the intersections of any part of $F$ with any part of $G$ differ by no more than one.
Suppose that $F$ and $G$ are uniform partitions of the finite set $\Omega$.

- $F \triangleright G$ means that $F$ is balanced with respect to $G$, in the sense that $N_{FG}N_{GF}$ is completely symmetric with non-zero off-diagonal elements, but $F$ is not strictly orthogonal to $G$.

- $F \triangleright\triangleright G$ means that $F \triangleright G$ and the relationship between $F$ and $G$ is binary or generalized binary, in the sense that the size of the intersections of any part of $F$ with any part of $G$ differ by no more than one.

- $F \bowtie G$ means that $F \triangleright G$ and $G \triangleright F$, which implies that $n_F = n_G$. 
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$. 
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $\mathbb{R}^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $\mathbb{R}^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.

Suppose that $R \perp C$, $R \perp L$ and $L \triangleright C$. 
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $\mathbb{R}^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.

Suppose that $R \perp C$, $R \perp L$ and $L \triangleright C$.

- Projecting onto $V_R^\perp$ leaves $V_C \cap V_0^\perp$ and $V_L \cap V_0^\perp$ unchanged, so the relation between $L$ and $C$ is unchanged.
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $\mathbb{R}^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.

Suppose that $R \perp C$, $R \perp L$ and $L \triangleright C$.

- Projecting onto $V_R^\perp$ leaves $V_C \cap V_0^\perp$ and $V_L \cap V_0^\perp$ unchanged, so the relation between $L$ and $C$ is unchanged.

- Projecting onto $V_L^\perp$ leaves $V_R \cap V_0^\perp$ unchanged and leaves $V_C \cap V_0^\perp$ inside $V_L + V_C$, which is orthogonal to $V_R \cap V_0^\perp$, so $R$ and $C$ have adjusted orthogonality with respect to $L$. 

More generally, given a set $F$ of partitions, if each $F$ in $F$ is non-orthogonal to at most one of the others then the pairwise relations suffice to describe the system.
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $R^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.

Suppose that $R \perp C$, $R \perp L$ and $L \succ C$.

- Projecting onto $V_R^\perp$ leaves $V_C \cap V_0^\perp$ and $V_L \cap V_0^\perp$ unchanged, so the relation between $L$ and $C$ is unchanged.

- Projecting onto $V_L^\perp$ leaves $V_R \cap V_0^\perp$ unchanged and leaves $V_C \cap V_0^\perp$ inside $V_L + V_C$, which is orthogonal to $V_R \cap V_0^\perp$, so $R$ and $C$ have adjusted orthogonality with respect to $L$. 

More generally, given a set $F$ of partitions, if each $F$ in $F$ is non-orthogonal to at most one of the others then the pairwise relations suffice to describe the system.
What about three partitions? Or more?

Let $R$, $C$ and $L$ be uniform partitions of $\Omega$.

If all three pairwise relations are orthogonality (possibly including refinement) then we get a nice decomposition of $\mathbb{R}^\Omega$ into orthogonal subspaces, and each pair has adjusted orthogonality with respect to the third.

Suppose that $R \perp C$, $R \perp L$ and $L \triangleright C$.

- Projecting onto $V_R^\perp$ leaves $V_C \cap V_0^\perp$ and $V_L \cap V_0^\perp$ unchanged, so the relation between $L$ and $C$ is unchanged.

- Projecting onto $V_L^\perp$ leaves $V_R \cap V_0^\perp$ unchanged and leaves $V_C \cap V_0^\perp$ inside $V_L + V_C$, which is orthogonal to $V_R \cap V_0^\perp$, so $R$ and $C$ have adjusted orthogonality with respect to $L$.

More generally, given a set $\mathcal{F}$ of partitions, if each $F$ in $\mathcal{F}$ is non-orthogonal to at most one of the others then the pairwise relations suffice to describe the system.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Suppose that we have 3 uniform partitions $R$, $C$ and $L$, and only one relation is not orthogonality.
Three partitions: only one non-orthogonality

Suppose that we have 3 uniform partitions $R$, $C$ and $L$, and only one relation is not orthogonality.

In the nicest case, the relation between $C$ and $L$ is balance in both directions.
Definition (Youden, 1937)
An $n \times m$ Youden square is a set of size $nm$ with uniform partitions into $n$ rows ($R$), $m$ columns ($C$) and $m$ letters ($L$) such that all pairwise relations are binary, $R \perp C$, $R \perp L$ and $L \bowtie C$. 

Example ($n = 3$ and $m = 7$)

\begin{array}{ccccccc}
A & B & C & D & E & F & G \\
B & D & F & E & G & A & C \\
C & F & E & A & B & G & D \\
\end{array}
Youden squares

Definition (Youden, 1937)

An $n \times m$ **Youden square** is a set of size $nm$ with uniform partitions into $n$ rows ($R$), $m$ columns ($C$) and $m$ letters ($L$) such that all pairwise relations are binary, $R \perp C$, $R \perp L$ and $L \bowtie C$.

Example ($n = 3$ and $m = 7$)

\[
\begin{array}{ccccccc}
A & B & C & D & E & F & G \\
B & D & F & E & G & A & C \\
C & F & E & A & B & G & D \\
\end{array}
\]
Youden squares

Definition (Youden, 1937)
An $n \times m$ Youden square is a set of size $nm$ with uniform partitions into $n$ rows ($R$), $m$ columns ($C$) and $m$ letters ($L$) such that all pairwise relations are binary, $R \perp C$, $R \perp L$ and $L \nmid C$.

Example ($n = 3$ and $m = 7$)

\[
\begin{array}{ccccccc}
A & B & C & D & E & F & G \\
B & D & F & E & G & A & C \\
C & F & E & A & B & G & D \\
\end{array}
\]

Theorem
Every symmetric balanced incomplete-block design can be arranged as a Youden square.
Youden squares

Definition (Youden, 1937)
An $n \times m$ Youden square is a set of size $nm$ with uniform partitions into $n$ rows ($R$), $m$ columns ($C$) and $m$ letters ($L$) such that all pairwise relations are binary, $R \perp C$, $R \perp L$ and $L \nparallel C$.

Example ($n = 3$ and $m = 7$)

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td>F</td>
<td>E</td>
<td>G</td>
<td>A</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>F</td>
<td>E</td>
<td>A</td>
<td>B</td>
<td>G</td>
<td>D</td>
<td></td>
</tr>
</tbody>
</table>

Theorem
Every symmetric balanced incomplete-block design can be arranged as a Youden square.

Proof.
Use Hall’s Marriage Theorem to sequentially choose the letters in each row as a set of distinct representatives.
Theorem
Suppose that $L$ and $B$ are uniform partitions with $n_L = n_B$ and $L \wedge B = E$. Then the elements of $\Omega$ can be arranged in a $k_B \times n_B$ rectangle such that the columns are the parts of $B$ and each letter occurs exactly once in each row.
Theorem
Suppose that $L$ and $B$ are uniform partitions with $n_L = n_B$ and $L \land B = E$. Then the elements of $\Omega$ can be arranged in a $k_B \times n_B$ rectangle such that the columns are the parts of $B$ and each letter occurs exactly once in each row.

Example (Not balanced)
Slightly more general theorem

**Theorem**
Suppose that \( L \) and \( B \) are uniform partitions with \( n_L = n_B \) and \( L \land B = E \). Then the elements of \( \Omega \) can be arranged in a \( k_B \times n_B \) rectangle such that the columns are the parts of \( B \) and each letter occurs exactly once in each row.

**Example (Not balanced)**

\[
\begin{array}{ccc}
A & D & G \\
B & E & H \\
C & F & I
\end{array}
\begin{array}{ccc}
A & B & C \\
D & E & F \\
G & H & I
\end{array}
\begin{array}{ccc}
A & B & C \\
E & F & D \\
I & G & H
\end{array}
\]
Theorem
Suppose that \( L \) and \( B \) are uniform partitions with \( n_L = n_B \) and \( L \cap B = E \). Then the elements of \( \Omega \) can be arranged in a \( k_B \times n_B \) rectangle such that the columns are the parts of \( B \) and each letter occurs exactly once in each row.

Example (Not balanced)

\[
\begin{array}{cccc}
A & D & G & A \\
B & E & H & B \\
C & F & I & C \\
\end{array}
\]

\[
\begin{array}{cccc}
A & B & C & A \\
D & E & F & B \\
G & H & I & C \\
\end{array}
\]

\[
\begin{array}{cccc}
A & D & H & G \\
B & I & E & F \\
C & C & C & C \\
\end{array}
\]
Theorem
Suppose that $L$ and $B$ are uniform partitions with $n_L = n_B$ and $L \wedge B = E$. Then the elements of $\Omega$ can be arranged in a $k_B \times n_B$ rectangle such that the columns are the parts of $B$ and each letter occurs exactly once in each row.

Example (Not balanced)

\[
\begin{array}{cccccc}
A & D & G & A & B & C \\
B & E & H & D & E & F \\
C & F & I & G & H & I \\
\end{array}
\]

\[
\begin{array}{cccccc}
A & D & H & A & B & C \\
C & E & G & E & F & D \\
B & F & I & I & G & H \\
\end{array}
\]

\[
\begin{array}{cccccc}
A & D & H & G & B & I \\
B & E & F & I & E & F \\
C & A & C & G & H & C \\
\end{array}
\]
Theorem

Suppose that \( L \) and \( B \) are uniform partitions with \( n_L = n_B \) and \( L \wedge B = E \). Then the elements of \( \Omega \) can be arranged in a \( k_B \times n_B \) rectangle such that the columns are the parts of \( B \) and each letter occurs exactly once in each row.

Example (Not balanced)

\[
\begin{array}{cccc}
  A & D & G & A \\
  B & E & H & B \\
  C & F & I & C \\
\end{array}
\]

\[
\begin{array}{cccc}
  A & D & H & G \\
  B & F & I & D \\
  C & A & G & H \\
\end{array}
\]
**Theorem**

Suppose that \( L \) and \( B \) are uniform partitions with \( n_L = n_B \) and \( L \land B = E \). Then the elements of \( \Omega \) can be arranged in a \( k_B \times n_B \) rectangle such that the columns are the parts of \( B \) and each letter occurs exactly once in each row.

**Example (Not balanced)**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>D</th>
<th>G</th>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>D</td>
<td>G</td>
<td></td>
<td></td>
<td>D</td>
<td>E</td>
<td>F</td>
<td></td>
<td>E</td>
<td>F</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td>E</td>
<td>I</td>
<td></td>
<td></td>
<td>G</td>
<td>H</td>
<td>I</td>
<td></td>
<td>I</td>
<td>G</td>
<td>H</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>D</th>
<th>H</th>
<th>G</th>
<th>B</th>
<th>I</th>
<th>E</th>
<th>F</th>
<th>C</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>F</td>
<td>I</td>
<td>D</td>
<td>E</td>
<td>C</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>E</td>
<td>G</td>
<td>A</td>
<td>H</td>
<td>F</td>
<td>I</td>
<td>B</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bailey

Relations among partitions
A symmetric incomplete-block design can be viewed as a regular bipartite graph. There is one vertex for each block, and one vertex for each letter. If letter $i$ is in block $j$ then there is an edge between vertex $i$ and vertex $j$. By Hall's Marriage Theorem, this graph has a matching (a set of edges including each vertex exactly once). We can use this matching to make the top row of the rectangle: columns are blocks, and the matching tells us what letter to put in each column. Remove those edges from the graph. This leaves a regular bipartite graph whose degree is one less than it was in the previous graph. Use induction on the degree. Degree 1 corresponds to a single matching, so the induction can start.
A symmetric incomplete-block design can be viewed as a regular bipartite graph. There is one vertex for each block, and one vertex for each letter. If letter $i$ is in block $j$ then there is an edge between vertex $i$ and vertex $j$.

By Hall’s Marriage Theorem, this graph has a matching (a set of edges including each vertex exactly once). We can use this matching to make the top row of the rectangle: columns are blocks, and the matching tells us what letter to put in each column.
A symmetric incomplete-block design can be viewed as a regular bipartite graph. There is one vertex for each block, and one vertex for each letter. If letter $i$ is in block $j$ then there is an edge between vertex $i$ and vertex $j$.

By Hall’s Marriage Theorem, this graph has a matching (a set of edges including each vertex exactly once). We can use this matching to make the top row of the rectangle: columns are blocks, and the matching tells us what letter to put in each column.

Remove those edges from the graph. This leaves a regular bipartite graph whose degree is one less than it was in the previous graph.
A symmetric incomplete-block design can be viewed as a regular bipartite graph. There is one vertex for each block, and one vertex for each letter. If letter \( i \) is in block \( j \) then there is an edge between vertex \( i \) and vertex \( j \).

By Hall’s Marriage Theorem, this graph has a matching (a set of edges including each vertex exactly once). We can use this matching to make the top row of the rectangle: columns are blocks, and the matching tells us what letter to put in each column.

Remove those edges from the graph. This leaves a regular bipartite graph whose degree is one less than it was in the previous graph.

Use induction on the degree.
A symmetric incomplete-block design can be viewed as a regular bipartite graph. There is one vertex for each block, and one vertex for each letter. If letter $i$ is in block $j$ then there is an edge between vertex $i$ and vertex $j$.

By Hall’s Marriage Theorem, this graph has a matching (a set of edges including each vertex exactly once). We can use this matching to make the top row of the rectangle: columns are blocks, and the matching tells us what letter to put in each column.

Remove those edges from the graph. This leaves a regular bipartite graph whose degree is one less than it was in the previous graph.

Use induction on the degree. Degree 1 corresponds to a single matching, so the induction can start.
Outline

- Three or four partitions with nice pairwise relations.
- Youden squares.
- Double Youden rectangles.
- Triple arrays.
Outline

▶ Three or four partitions with nice pairwise relations.
▶ Youden squares.
▶ Double Youden rectangles.
▶ Triple arrays.
If we have 4 uniform partitions $R, C, L$ and $G$, we could have two disjoint pairs related by $\bowtie$. 
If we have 4 uniform partitions $R, C, L$ and $G$, we could have two disjoint pairs related by $\bowtie$.

$$R \bowtie G \quad n \text{ parts of size } m$$

everything above is strictly orthogonal to everything below

$$C \bowtie L \quad m \text{ parts of size } n$$
Double Youden rectangles

Definition (Bailey, 1989)

An \( n \times m \) **double Youden rectangle** is a set of size \( nm \) with uniform partitions into \( n \) rows (\( R \)), \( m \) columns (\( C \)), \( m \) Latin letters (\( L \)) and \( n \) Greek letters (\( G \)) such that all pairwise relations (apart from that between \( R \) and \( G \)) are binary, \( R \perp C, R \perp L, G \perp C, G \perp L, L \bowtie C \) and \( R \bowtie G \).
Double Youden rectangles

Definition (Bailey, 1989)

An $n \times m$ **double Youden rectangle** is a set of size $nm$ with uniform partitions into $n$ rows ($R$), $m$ columns ($C$), $m$ Latin letters ($L$) and $n$ Greek letters ($G$) such that all pairwise relations (apart from that between $R$ and $G$) are binary, $R \perp C$, $R \perp L$, $G \perp C$, $G \perp L$, $L \bowtie C$ and $R \bowtie G$.

Example ($n = 4$ and $m = 13$, Preece (1982))
Donald Preece, a statistician at Rothamsted Experimental Station, discovered this design in the 1980s. He was so delighted by his discovery that he made this picture by sticking real playing cards onto a cardboard background. This was hung up in the Statistics Department.
A picture made from playing cards

Donald Preece, a statistician at Rothamsted Experimental Station, discovered this design in the 1980s. He was so delighted by his discovery that he made this picture by sticking real playing cards onto a cardboard background. This was hung up in the Statistics Department.
This photograph was taken about 30 years after that discovery.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Three or four partitions with nice pairwise relations.
Youden squares.
Double Youden rectangles.
Triple arrays.
Suppose that we have 3 uniform partitions $R$, $C$ and $L$, and only one relation is orthogonality.

![Diagram of three partitions: $R$, $C$, and $L$ with orthogonality indicated by a perpendicular symbol.]
Suppose that we have 3 uniform partitions $R$, $C$ and $L$, and only one relation is orthogonality.

We should like

- $R$ and $C$ to have adjusted orthogonality with respect to $L$;
- the relation between $R$ and $L$ to be “nice”;
- the relation between $C$ and $L$ to be “nice”.

We have

$L \perp R \perp C$
How can this be nice?

We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$. 
We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$. 

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$. 

Let $z_R$ be a non-zero vector in $V_R \cap V_0^\perp$. 

Let $z_C$ be a non-zero vector in $V_C \cap V_0^\perp$. 

$x_R = z_R + y_R$ $x_C = z_C + y_C$ because $R \perp C$.

$y_R \perp y_C$ by adjusted orthogonality.

$z_R \perp z_C$ by construction.

So $P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp)$.
We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$.
Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$. 

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$.
Put $z_C = P_L(x_C)$ and $y_C = x_C - z_C$. 

$x_R = z_R + y_R$ 
$x_C = z_C + y_C$ 
because $R \perp C$ 
y_R \perp y_C$ by adjusted orthogonality 
y_i \perp z_j by construction 
so $z_R \perp z_C$. 
Hence $P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp)$. 

We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$. Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$.

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$. 

$z_R \perp z_C$ by construction.

$y_R \perp y_C$ by adjusted orthogonality.

Hence $P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp)$. 

$z_R \perp z_C$. 

$y_R \perp y_C$. 

$y_i \perp z_j$ by construction.

$\mathbf{Bailey}$
We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$. Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$.

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$. Put $z_C = P_L(x_C)$ and $y_C = x_C - z_C$. 
How can this be nice?

We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$.

Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$.

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$.

Put $z_C = P_L(x_C)$ and $y_C = x_C - z_C$.

\[
  x_R = z_R + y_R \quad x_C = z_C + y_C
\]
How can this be nice?

We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$.
Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$.

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$.
Put $z_C = P_L(x_C)$ and $y_C = x_C - z_C$.

$$x_R = z_R + y_R \quad x_C = z_C + y_C$$

$$x_R \perp x_C \quad \text{because } R \perp C$$
$$y_R \perp y_C \quad \text{by adjusted orthogonality}$$
$$y_i \perp z_j \quad \text{by construction}$$
We assume that \( R \perp C \) and that \( R \) and \( C \) have adjusted orthogonality with respect to \( L \).

Let \( x_R \) be a non-zero vector in \( V_R \cap V_0^\perp \).
Put \( z_R = P_L(x_R) \) and \( y_R = x_R - z_R \).

Let \( x_C \) be a non-zero vector in \( V_C \cap V_0^\perp \).
Put \( z_C = P_L(x_C) \) and \( y_C = x_C - z_C \).

\[
\begin{align*}
x_R &= z_R + y_R \\
x_C &= z_C + y_C
\end{align*}
\]

\( x_R \perp x_C \) because \( R \perp C \)
\( y_R \perp y_C \) by adjusted orthogonality
\( y_i \perp z_j \) by construction

\[
\begin{cases}
x_R \perp x_C \\
y_R \perp y_C \\
y_i \perp z_j
\end{cases}
\]

so \( z_R \perp z_C \).
We assume that $R \perp C$ and that $R$ and $C$ have adjusted orthogonality with respect to $L$.

Let $x_R$ be a non-zero vector in $V_R \cap V_0^\perp$.
Put $z_R = P_L(x_R)$ and $y_R = x_R - z_R$.

Let $x_C$ be a non-zero vector in $V_C \cap V_0^\perp$.
Put $z_C = P_L(x_C)$ and $y_C = x_C - z_C$.

$$x_R = z_R + y_R \quad x_C = z_C + y_C$$

$x_R \perp x_C$ because $R \perp C$
$y_R \perp y_C$ by adjusted orthogonality
$y_i \perp z_j$ by construction

Hence $P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp)$. 
What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

\[ P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp). \]  \hspace{1cm} (1)

If $C \trianglelefteq L$ then $\dim(P_L(V_C \cap V_0^\perp)) = n_C - 1$. If $R \trianglelefteq L$ then $\dim(P_L(V_R \cap V_0^\perp)) = n_R - 1$. If $C \trianglelefteq L$ and $R \trianglelefteq L$ then Equation (1) forces

\[ (n_C - 1) + (n_R - 1) \leq n_L - 1. \]
What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

$$P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp).$$  (1)

If $L \triangleright C$ then $P_L(V_C \cap V_0^\perp) = V_L \cap V_0^\perp$

(the proof is similar to the proof of Fisher’s Inequality) and so Equation (1) is impossible.
What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

\[ P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp). \]  

(1)

If $L \triangleright C$ then $P_L(V_C \cap V_0^\perp) = V_L \cap V_0^\perp$ (the proof is similar to the proof of Fisher’s Inequality) and so Equation (1) is impossible.

So the only nice case that we can consider is $C \triangleright L$ and $R \triangleright L$. 

What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

$$P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp).$$  (1)

If $L \triangleright C$ then $P_L(V_C \cap V_0^\perp) = V_L \cap V_0^\perp$ (the proof is similar to the proof of Fisher’s Inequality) and so Equation (1) is impossible.

So the only nice case that we can consider is $C \triangleright L$ and $R \triangleright L$.

If $C \triangleright L$ then $\dim (P_L(V_C \cap V_0^\perp)) = n_C - 1$. 
What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

$$P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp).$$  \hfill (1)

If $L \triangleright C$ then $P_L(V_C \cap V_0^\perp) = V_L \cap V_0^\perp$

(the proof is similar to the proof of Fisher’s Inequality)

and so Equation (1) is impossible.

So the only nice case that we can consider is $C \triangleright L$ and $R \triangleright L$.

If $C \triangleright L$ then $\dim (P_L(V_C \cap V_0^\perp)) = n_C - 1$.

If $R \triangleright L$ then $\dim (P_L(V_R \cap V_0^\perp)) = n_R - 1$. 

What sort of balance can we assume?

We have shown that, if $R \perp C$ and $R$ and $C$ have adjusted orthogonality with respect to $L$, then

$$P_L(V_R \cap V_0^\perp) \perp P_L(V_C \cap V_0^\perp). \quad (1)$$

If $L \triangleright C$ then $P_L(V_C \cap V_0^\perp) = V_L \cap V_0^\perp$

(the proof is similar to the proof of Fisher’s Inequality) and so Equation (1) is impossible.

So the only nice case that we can consider is $C \triangleright L$ and $R \triangleright L$.

If $C \triangleright L$ then $\dim (P_L(V_C \cap V_0^\perp)) = n_C - 1$.

If $R \triangleright L$ then $\dim (P_L(V_R \cap V_0^\perp)) = n_R - 1$.

If $C \triangleright L$ and $R \triangleright L$ then Equation (1) forces

$$(n_C - 1) + (n_R - 1) \leq n_L - 1.$$
**Triple arrays**

**Definition (McSorley, Phillips, Wallis and Yucas, 2005)**

An $r \times c$ rectangle with one of $v$ letters allocated to each cell is an **triple array** if all partitions are uniform, all pairwise relations are binary, $R \bot C$, $R \triangleright L$, $C \triangleright L$ and $R$ and $C$ have adjusted orthogonality with respect to $L$. 

So $n_R = r = k_C$, $n_C = c = k_R$, $n_L = v$ and $k_L = rc/v$. 

Also, every pair of rows have the same number of letters in common, every pair of columns have the same number of letters in common, and every row has $k_L$ letters in common with every column. These are among the designs discussed by Preece (1966) and Agrawal (1966).
Definition (McSorley, Phillips, Wallis and Yucas, 2005)
An $r \times c$ rectangle with one of $v$ letters allocated to each cell is an **triple array** if all partitions are uniform, all pairwise relations are binary, $R \perp C$, $R \triangleright L$, $C \triangleright L$ and $R$ and $C$ have adjusted orthogonality with respect to $L$.

So $n_R = r = k_C$, $n_C = c = k_R$, $n_L = v$ and $k_L = rc/v$. 
Definition (McSorley, Phillips, Wallis and Yucas, 2005)

An $r \times c$ rectangle with one of $v$ letters allocated to each cell is an **triple array** if all partitions are uniform, all pairwise relations are binary, $R \perp C$, $R \triangleright L$, $C \triangleright L$ and $R$ and $C$ have adjusted orthogonality with respect to $L$.

So $n_R = r = k_C$, $n_C = c = k_R$, $n_L = v$ and $k_L = rc/v$.

Also, every pair of rows have the same number of letters in common, every pair of columns have the same number of letters in common, and every row has $k_L$ letters in common with every column.
Definition (McSorley, Phillips, Wallis and Yucas, 2005)

An $r \times c$ rectangle with one of $v$ letters allocated to each cell is an **triple array** if all partitions are uniform, all pairwise relations are binary, $R \perp C$, $R \triangleright L$, $C \triangleright L$, and $R$ and $C$ have adjusted orthogonality with respect to $L$.

So $n_R = r = k_C$, $n_C = c = k_R$, $n_L = v$ and $k_L = rc/v$.

Also, every pair of rows have the same number of letters in common, every pair of columns have the same number of letters in common, and every row has $k_L$ letters in common with every column.

These are among the designs discussed by Preece (1966) and Agrawal (1966).
Extremal triple arrays

Theorem (Bagchi, 1998)

If a triple array has $r$ rows, $c$ columns and $v$ letters then

$v \geq r + c - 1$. 

Definition

A triple array is extremal if $v = r + c - 1$. 

Given an extremal triple array, the following construction gives a symmetric balanced incomplete-block design (SBIBD) for $r + c$ points in blocks of size $r$.

1. The points are the (names of the) rows and columns.
2. Each letter gives a block, consisting of the columns in which it occurs and the rows in which it does not occur.
3. The final block contains (the names of) all the rows.
Theorem (Bagchi, 1998)

If a triple array has $r$ rows, $c$ columns and $v$ letters then

$$v \geq r + c - 1.$$ 

Definition

A triple array is extremal if $v = r + c - 1$. 
Theorem (Bagchi, 1998)

If a triple array has $r$ rows, $c$ columns and $v$ letters then

$v \geq r + c - 1$.

Definition

A triple array is **extremal** if $v = r + c - 1$.

Given an extremal triple array, the following construction gives a symmetric balanced incomplete-block design (SBIBD) for $r + c$ points in blocks of size $r$.

1. The points are the (names of the) rows and columns.
Extremal triple arrays

Theorem (Bagchi, 1998)
If a triple array has \( r \) rows, \( c \) columns and \( v \) letters then
\[ v \geq r + c - 1. \]

Definition
A triple array is extremal if
\[ v = r + c - 1. \]

Given an extremal triple array, the following construction gives
a symmetric balanced incomplete-block design (SBIBD) for
\( r + c \) points in blocks of size \( r \).

1. The points are the (names of the) rows and columns.
2. Each letter gives a block, consisting of the columns in
which it occurs and the rows in which it does not occur.
Theorem (Bagchi, 1998)

If a triple array has $r$ rows, $c$ columns and $v$ letters then $v \geq r + c - 1$.

Definition

A triple array is extremal if $v = r + c - 1$.

Given an extremal triple array, the following construction gives a symmetric balanced incomplete-block design (SBIBD) for $r + c$ points in blocks of size $r$.

1. The points are the (names of the) rows and columns.
2. Each letter gives a block, consisting of the columns in which it occurs and the rows in which it does not occur.
3. The final block contains (the names of) all the rows.
An extremal triple array with \( r = 5 \), \( c = 6 \) and \( v = 10 \)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>B</td>
<td>A</td>
<td>E</td>
<td>D</td>
<td>J</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>G</td>
<td>H</td>
<td>B</td>
<td>I</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>9</td>
<td>J</td>
<td>I</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>J</td>
<td>H</td>
<td>C</td>
<td>E</td>
<td>I</td>
</tr>
<tr>
<td>3</td>
<td>H</td>
<td>D</td>
<td>C</td>
<td>F</td>
<td>G</td>
<td>A</td>
</tr>
</tbody>
</table>

An \( r \times c \) rectangle, each cell containing one of \( r + c - 1 \) letters, such that

- rows \( R \) are strictly orthogonal to columns \( C \), with all intersections of size 1;
- rows are balanced with respect to letters (\( L \)) (every pair of rows has the same number (3) of letters in common);
- columns are balanced with respect to letters;
- rows and columns have adjusted orthogonality with respect to \( L \) (the set of letters in each row has constant size of intersection with the set of letters in each column).
### Triple array to SBIBD

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>B</td>
<td>A</td>
<td>E</td>
<td>D</td>
<td>J</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>G</td>
<td>H</td>
<td>B</td>
<td>I</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>9</td>
<td>J</td>
<td>I</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>J</td>
<td>H</td>
<td>C</td>
<td>E</td>
<td>I</td>
</tr>
<tr>
<td>3</td>
<td>H</td>
<td>D</td>
<td>C</td>
<td>F</td>
<td>G</td>
<td>A</td>
</tr>
</tbody>
</table>

- The points are 1, 4, 9, 5, 3, 0, 2, 6, 7, 8, X.
The points are 1, 4, 9, 5, 3, 0, 2, 6, 7, 8, X.
Block A contains points 2, 6, X, 4, 5.
The points are 1, 4, 9, 5, 3, 0, 2, 6, 7, 8, X.

Block A contains points 2, 6, X, 4, 5.

And so on.
The points are 1, 4, 9, 5, 3, 0, 2, 6, 7, 8, X.
Block A contains points 2, 6, X, 4, 5.
And so on.
Block J contains points 0, 2, 8, 4, 3.
The points are 1, 4, 9, 5, 3, 0, 2, 6, 7, 8, X.

Block A contains points 2, 6, X, 4, 5.

And so on.

Block J contains points 0, 2, 8, 4, 3.

The final block contains points 1, 4, 9, 5, 3.
Start with a SBIBD: can we construct the triple array?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Start with a SBIBD: can we construct the triple array?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**column name is in**

A B D E F J
B D E G H I
C E F H I J
A C D F F H

**row name is not in**

A B C G I J
C E F H I J
A C D F F H

Bailey Relations among partitions
Start with a SBIBD: can we construct the triple array?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>column name</th>
<th>0</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>rowname is not in</th>
<th>A</th>
<th>B</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>D</td>
<td>E</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>G</td>
<td>I</td>
<td>J</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>E</td>
<td>F</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>D</td>
<td>F</td>
<td>F</td>
<td>H</td>
<td></td>
</tr>
</tbody>
</table>

Put one letter in each cell and obtain these subsets in rows and columns.

Bailey Relations among partitions
Start with a SBIBD: can we construct the triple array?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>column name is in</th>
<th>B</th>
<th>A</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F</td>
<td>D</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>H</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td></td>
<td>H</td>
<td>I</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td></td>
<td>J</td>
<td>J</td>
<td>H</td>
<td>I</td>
<td>I</td>
<td>I</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>row name is not in</th>
<th>A</th>
<th>B</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B</td>
<td>D</td>
<td>E</td>
<td>G</td>
<td>H</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>G</td>
<td>I</td>
<td>J</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>E</td>
<td>F</td>
<td>H</td>
<td>I</td>
<td>J</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>C</td>
<td>D</td>
<td>F</td>
<td>F</td>
<td>H</td>
</tr>
</tbody>
</table>

Put one letter in each cell and obtain these subsets in rows and columns.
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column? Fon-der-Flaass, 1997: the general problem is NP-complete.
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

▶ Not if the allowable subsets have size ≤ 2.
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size \( \leq 2 \).
- Agrawal (1966): if \( k_L > 2 \) then it was “always possible in the examples tried by the author”.

Seberry (1979); Street (1981); Bailey and Heidtmann (1994); Bagchi (1998); Preece, Wallis and Yucas (2005) gave explicit constructions for \( q \times (q + 1) \) when \( q \) is an odd prime power and \( q > 3 \).

Computer search always gives a positive result if \( k_L > 2 \).

Your task: Proof or counter-example.
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size \( \leq 2 \).
- Agrawal (1966): if \( k_L > 2 \) then it was “always possible in the examples tried by the author”.
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size \( \leq 2 \).
- Agrawal (1966): if \( k_L > 2 \) then it was “always possible in the examples tried by the author”.
- Seberry (1979); Street (1981); Bailey and Heidtmann (1994); Bagchi (1998); Preece, Wallis and Yucas (2005) gave explicit constructions for \( q \times (q + 1) \) when \( q \) is an odd prime power and \( q > 3 \).
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size $\leq 2$.
- Agrawal (1966): if $k_L > 2$ then it was “always possible in the examples tried by the author”.
- Seberry (1979); Street (1981); Bailey and Heidtmann (1994); Bagchi (1998); Preece, Wallis and Yucas (2005) gave explicit constructions for $q \times (q + 1)$ when $q$ is an odd prime power and $q > 3$.
- Computer search always gives a positive result if $k_L > 2$. 
Problem: can you do it?

Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der-Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size \( \leq 2 \).
- Agrawal (1966): if \( k_L > 2 \) then it was “always possible in the examples tried by the author”.
- Seberry (1979); Street (1981); Bailey and Heidtmann (1994); Bagchi (1998); Preece, Wallis and Yucas (2005) gave explicit constructions for \( q \times (q + 1) \) when \( q \) is an odd prime power and \( q > 3 \).
- Computer search always gives a positive result if \( k_L > 2 \).
Given a subset of letters allowed for each cell, is it possible to choose an array of distinct representatives, one per cell, so that no letter is repeated in a row or column?

Fon-der- Flaass, 1997: the general problem is NP-complete.

Suppose the allowable subsets come from an SBIBD in the way that I showed?

- Not if the allowable subsets have size \( \leq 2 \).
- Agrawal (1966): if \( k_L > 2 \) then it was “always possible in the examples tried by the author”.
- Seberry (1979); Street (1981); Bailey and Heidtmann (1994); Bagchi (1998); Preece, Wallis and Yucas (2005) gave explicit constructions for \( q \times (q + 1) \) when \( q \) is an odd prime power and \( q > 3 \).
- Computer search always gives a positive result if \( k_L > 2 \).

**Your task:** Proof or counter-example.