Laplacian eigenvalues and optimality: IV. Further topics

R. A. Bailey and Peter J. Cameron
Groups and Graphs, Designs and Dynamics
Yichang, China, August 2019

Further topics

The last lecture will discuss some additional topics. These include:

- Sylvester designs (an interesting class of examples);

Further topics

The last lecture will discuss some additional topics. These include:

- Sylvester designs (an interesting class of examples);
- how to recognise the concurrence graph of a block design;

Further topics

The last lecture will discuss some additional topics. These include:

- Sylvester designs (an interesting class of examples);
- how to recognise the concurrence graph of a block design;
- variance-balanced designs;

Further topics

The last lecture will discuss some additional topics. These include:

- Sylvester designs (an interesting class of examples);
- how to recognise the concurrence graph of a block design;
- variance-balanced designs;
- the relation of optimality parameters to other graph invariants such as the Tutte polynomial.

Block designs and concurrence graphs

We have seen that the values of the various parameters associated with optimality criteria of block designs depend only on the concurrence graph of the design: to find the optimal design we have to find the graph which maximizes the number of spanning trees, or minimizes the average resistance; or whatever.

Block designs and concurrence graphs

We have seen that the values of the various parameters associated with optimality criteria of block designs depend only on the concurrence graph of the design: to find the optimal design we have to find the graph which maximizes the number of spanning trees, or minimizes the average resistance; or whatever.

For block designs with block size 2, the design is the same as its concurrence graph (treatments are vertices and blocks are edges). But for larger block size, there are interesting questions.

Sylvester designs

There is no affine plane, or even pair of orthogonal Latin squares, of order 6. As a substitute, we propose the Sylvester designs, which I now describe.

Sylvester designs

There is no affine plane, or even pair of orthogonal Latin squares, of order 6. As a substitute, we propose the Sylvester designs, which I now describe.

These designs have 36 points and 48 blocks of size 6. Two points are contained in either one or two blocks: the pairs lying in two blocks are the edges of the Sylvester graph, to be defined. So the concurrence matrix has 8 on the diagonal, and 1 or 2 off-diagonal, with 2 for edges of the Sylvester graph. Thus the concurrence matrix is $8I + (J - I + A)$, where A is the adjacency matrix of the Sylvester graph.

Sylvester designs

There is no affine plane, or even pair of orthogonal Latin squares, of order 6. As a substitute, we propose the Sylvester designs, which I now describe.

These designs have 36 points and 48 blocks of size 6. Two points are contained in either one or two blocks: the pairs lying in two blocks are the edges of the Sylvester graph, to be defined. So the concurrence matrix has 8 on the diagonal, and 1 or 2 off-diagonal, with 2 for edges of the Sylvester graph. Thus the concurrence matrix is $8I + (J - I + A)$, where A is the adjacency matrix of the Sylvester graph.

As we have seen, designs with this concurrence matrix will all have the same Laplacian eigenvalues, and so will coincide on the A-, D- and E-criteria.

Sylvester designs

There is no affine plane, or even pair of orthogonal Latin squares, of order 6. As a substitute, we propose the Sylvester designs, which I now describe.

These designs have 36 points and 48 blocks of size 6. Two points are contained in either one or two blocks: the pairs lying in two blocks are the edges of the Sylvester graph, to be defined. So the concurrence matrix has 8 on the diagonal, and 1 or 2 off-diagonal, with 2 for edges of the Sylvester graph. Thus the concurrence matrix is $8I + (J - I + A)$, where A is the adjacency matrix of the Sylvester graph.

As we have seen, designs with this concurrence matrix will all have the same Laplacian eigenvalues, and so will coincide on the A-, D- and E-criteria.

Conjecture

Sylvester designs are A-, D- and E-optimal among all block designs with 36 *points and* 48 *blocks of size* 6*.*

The outer automorphism of S_6

The symmetric group S_6 has an outer automorphism. This means that it acts in two different ways on sets of six points, say A and B. An element of S_6 which is a transposition on A is a product of three transpositions on B.

The outer automorphism of S_6

The symmetric group S_6 has an outer automorphism. This means that it acts in two different ways on sets of six points, say A and B. An element of S_6 which is a transposition on A is a product of three transpositions on B.

The set B can be constructed as the set of 1-factorisations of the complete graph on six vertices (on the vertex set A). The details follow on the next slide.

The outer automorphism of S_6

The symmetric group S_6 has an outer automorphism. This means that it acts in two different ways on sets of six points, say A and B. An element of S_6 which is a transposition on A is a product of three transpositions on B.

The set B can be constructed as the set of 1-factorisations of the complete graph on six vertices (on the vertex set A). The details follow on the next slide.

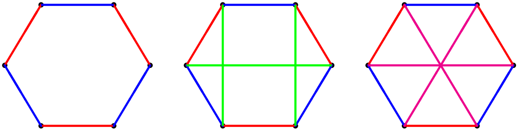
Now we define a graph on the vertex set $A \times B$ (the Cartesian product) by the rule that (a_1, b_1) is joined to (a_2, b_2) if and only if the transposition (a_1, a_2) on A corresponds to a product of three transpositions on B, one of which is (b_1, b_2). This is the Sylvester graph.

The outer automorphism of S_6

The group S_6 acts on a set A of six points. It also acts on the 15 2-subsets of A (edges of the complete graph, or duads in Sylvester's terminology), and on the $15 \cdot 6 \cdot 1/3! = 15$ partitions into three sets of two (1-factors, or Sylvester's synthemes). The set B of size 6 is the set of partitions of the duads into five synthemes (1-factorisations, or Sylvester's synthematic totals.

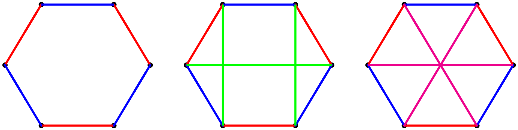
The outer automorphism of S_6

The group S_6 acts on a set A of six points. It also acts on the 15 2-subsets of A (edges of the complete graph, or duads in Sylvester's terminology), and on the $15 \cdot 6 \cdot 1/3! = 15$ partitions into three sets of two (1-factors, or Sylvester's synthemes). The set B of size 6 is the set of partitions of the duads into five synthemes (1-factorisations, or Sylvester's synthematic totals.



The outer automorphism of S_6

The group S_6 acts on a set A of six points. It also acts on the 15 2-subsets of A (edges of the complete graph, or duads in Sylvester's terminology), and on the $15 \cdot 6 \cdot 1/3! = 15$ partitions into three sets of two (1-factors, or Sylvester's synthemes). The set B of size 6 is the set of partitions of the duads into five synthemes (1-factorisations, or Sylvester's synthematic totals.



The first two synthemes in a total must form a 6-cycle. The remaining three must use the three long and six short diagonals. There are only two patterns for a syntheme of diagonals, shown in magenta and green. We cannot use the magenta since only short diagonals would remain. So the remaining three synthemes each consist of a long diagonal and two perpendicular short diagonals.

This shows that the synthematic total (partition of duads into synthemes) is unique up to isomorphism.

This shows that the synthematic total (partition of duads into synthemes) is unique up to isomorphism.
There are 15 choices of the first syntheme, 8 of the second, and $3 \cdot 2 \cdot 1$ of the remaining ones. Since the synthemes can be chosen in any order, the number of synthematic totals is $15 \cdot 8 \cdot 2!/5! = 6$.

This shows that the synthematic total (partition of duads into synthemes) is unique up to isomorphism.
There are 15 choices of the first syntheme, 8 of the second, and $3 \cdot 2 \cdot 1$ of the remaining ones. Since the synthemes can be chosen in any order, the number of synthematic totals is $15 \cdot 8 \cdot 2!/5! = 6$.
Let B be the set of six synthematic totals. Then the group S_6 acts on B, and it is easy to see that this action is not isomorphic to the action on the set A of vertices; so there is an outer automorphism mapping the first to the second. Moreover, we can reverse the procedure, and find that the square of this automorphism is inner.

This shows that the synthematic total (partition of duads into synthemes) is unique up to isomorphism.
There are 15 choices of the first syntheme, 8 of the second, and $3 \cdot 2 \cdot 1$ of the remaining ones. Since the synthemes can be chosen in any order, the number of synthematic totals is $15 \cdot 8 \cdot 2!/5! = 6$.
Let B be the set of six synthematic totals. Then the group S_6 acts on B, and it is easy to see that this action is not isomorphic to the action on the set A of vertices; so there is an outer automorphism mapping the first to the second. Moreover, we can reverse the procedure, and find that the square of this automorphism is inner.
It is remarkable that 6 is the only number n, finite or infinite, for which the symmetric group S_n has an outer automorphism.

The Sylvester graph

An alternative definition of the Sylvester graph: the vertex set is $A \times B$; the pairs (a_1, b_1) and (a_2, b_2) are joined if the duad $\{a_1, a_2\}$ belongs to the unique syntheme which the totals b_1 and b_2 have in common.

The Sylvester graph

An alternative definition of the Sylvester graph: the vertex set is $A \times B$; the pairs (a_1, b_1) and (a_2, b_2) are joined if the duad $\{a_1, a_2\}$ belongs to the unique syntheme which the totals b_1 and b_2 have in common.

The Sylvester graph is a distance-transitive graph on 36 vertices with valency 5. Its adjacency matrix has eigenvalues 5 (with multiplicity 1), 2 (multiplicity 16), -1 (multiplicity 10) and -3 (multiplicity 9). From these, the Laplacian eigenvalues of the concurrence matrix are easily computed: the non-trivial ones are 39, 42 and 44.

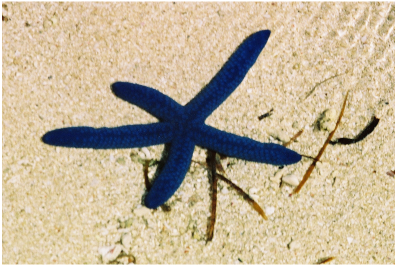
The Sylvester graph

An alternative definition of the Sylvester graph: the vertex set is $A \times B$; the pairs (a_1, b_1) and (a_2, b_2) are joined if the duad $\{a_1, a_2\}$ belongs to the unique syntheme which the totals b_1 and b_2 have in common.

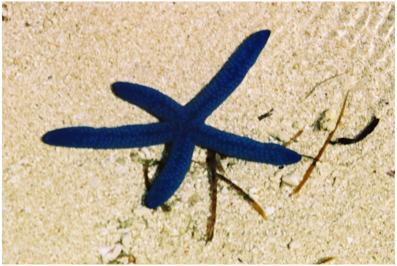
The Sylvester graph is a distance-transitive graph on 36 vertices with valency 5. Its adjacency matrix has eigenvalues 5 (with multiplicity 1), 2 (multiplicity 16), -1 (multiplicity 10) and -3 (multiplicity 9). From these, the Laplacian eigenvalues of the concurrence matrix are easily computed: the non-trivial ones are 39, 42 and 44.

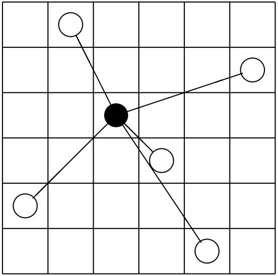
Its vertices can be regarded as the points of the 6×6 grid $A \times B$. A vertex and its five neighbours lie in distinct rows and columns. The graph contains no triangles or quadrangles. Any two vertices in different rows and columns lie at distance 1 or 2; if they are not adjacent, they have just one common neighbour.

Starfish

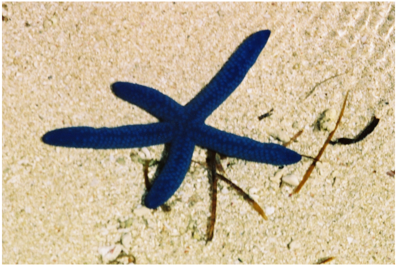


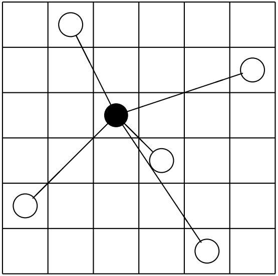
Starfish





Starfish





We define a starfish to consist of a vertex and its neighbours; a galaxy of starfish is the set of six starfish derived from the vertices in a column of the array.

A Sylvester design

Consider the following 48 sets of size 6:

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;
- the 36 starfish.

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;
- the 36 starfish.

It follows from the properties of the graph that two adjacent vertices lie in two blocks (the two starfish defined by these vertices), while any other pair of vertices are contained in a single block.

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;
- the 36 starfish.

It follows from the properties of the graph that two adjacent vertices lie in two blocks (the two starfish defined by these vertices), while any other pair of vertices are contained in a single block.

This design admits the full automorphism group of the graph, which is the automorphism group of S_6 (a group of order 1440).

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;
- the 36 starfish.

It follows from the properties of the graph that two adjacent vertices lie in two blocks (the two starfish defined by these vertices), while any other pair of vertices are contained in a single block.

This design admits the full automorphism group of the graph, which is the automorphism group of S_6 (a group of order 1440).

The design is resolvable, that is, the blocks can be partitioned into eight sets of six, each of which covers all the points. (The resolution classes are the rows, the columns, and the six galaxies of starfish.)

A Sylvester design

Consider the following 48 sets of size 6:

- the six rows and the six columns of the array;
- the 36 starfish.

It follows from the properties of the graph that two adjacent vertices lie in two blocks (the two starfish defined by these vertices), while any other pair of vertices are contained in a single block.
This design admits the full automorphism group of the graph, which is the automorphism group of S_6 (a group of order 1440).
The design is resolvable, that is, the blocks can be partitioned into eight sets of six, each of which covers all the points. (The resolution classes are the rows, the columns, and the six galaxies of starfish.)
Good designs with lower replication can be obtained by using only some galaxies.

More Sylvester designs

As well as this beautiful design, there are others.

More Sylvester designs

As well as this beautiful design, there are others.
Emlyn Williams discovered one using his CycDesign software.
It turned out to have the same A-value as the previous one, and a short computation showed that in fact it had the same concurrence matrix. However, the automorphism group of this design is the trivial group.

More Sylvester designs

As well as this beautiful design, there are others.
Emlyn Williams discovered one using his CycDesign software.
It turned out to have the same A-value as the previous one, and a short computation showed that in fact it had the same concurrence matrix. However, the automorphism group of this design is the trivial group.
Another Sylvester design, with 144 automorphisms, was found by Leonard Soicher using semi-Latin squares.

More Sylvester designs

As well as this beautiful design, there are others.
Emlyn Williams discovered one using his CycDesign software.
It turned out to have the same A-value as the previous one, and a short computation showed that in fact it had the same concurrence matrix. However, the automorphism group of this design is the trivial group.
Another Sylvester design, with 144 automorphisms, was found by Leonard Soicher using semi-Latin squares.
All these designs are resolvable.

More Sylvester designs

As well as this beautiful design, there are others.
Emlyn Williams discovered one using his CycDesign software.
It turned out to have the same A-value as the previous one, and a short computation showed that in fact it had the same concurrence matrix. However, the automorphism group of this design is the trivial group.
Another Sylvester design, with 144 automorphisms, was found by Leonard Soicher using semi-Latin squares.
All these designs are resolvable.

Problem

Can the Sylvester designs be classified up to isomorphism?

More Sylvester designs

As well as this beautiful design, there are others.
Emlyn Williams discovered one using his CycDesign software.
It turned out to have the same A-value as the previous one, and a short computation showed that in fact it had the same concurrence matrix. However, the automorphism group of this design is the trivial group.
Another Sylvester design, with 144 automorphisms, was found by Leonard Soicher using semi-Latin squares.
All these designs are resolvable.

Problem

Can the Sylvester designs be classified up to isomorphism?

This would probably be a very big computation!

Sparse versus dense

We have seen that the optimality criteria for block designs tend to agree on designs with dense concurrence graphs, but give very different results in the case where the concurrence graph is sparse.

Sparse versus dense

We have seen that the optimality criteria for block designs tend to agree on designs with dense concurrence graphs, but give very different results in the case where the concurrence graph is sparse.
We have also seen that optimality for block designs tends to agree with desirable characteristics for networks.

Sparse versus dense

We have seen that the optimality criteria for block designs tend to agree on designs with dense concurrence graphs, but give very different results in the case where the concurrence graph is sparse.
We have also seen that optimality for block designs tends to agree with desirable characteristics for networks.
Now sparse networks occur for the same reason as block designs with low replication, namely resource limitations. So these results are potentially of interest in network theory as well.

BIBDs

Recall that a BIBD for v treatments, with b blocks of size k, has the property that the replication of any treatment is a constant, r, and the concurrence of two treatments is a constant, λ, where

BIBDs

Recall that a BIBD for v treatments, with b blocks of size k, has the property that the replication of any treatment is a constant, r, and the concurrence of two treatments is a constant, λ, where

- $bk = vr$;

BIBDs

Recall that a BIBD for v treatments, with b blocks of size k, has the property that the replication of any treatment is a constant, r, and the concurrence of two treatments is a constant, λ, where

- $bk = vr$;
- $r(k-1) = (v-1)\lambda$.

BIBDs

Recall that a BIBD for v treatments, with b blocks of size k, has the property that the replication of any treatment is a constant, r, and the concurrence of two treatments is a constant, λ, where

- $bk = vr$;
- $r(k-1) = (v-1)\lambda$.

The concurrence graph of such a design is the λ-fold complete graph in which any two vertices are joined by λ edges. Moreover, the design is binary.

Steiner triple systems

For $k = 3$ and $\lambda = 1$, such a design is a Steiner triple system. The blocks are 3-subsets of the set of points, and two distinct points lie in a unique block.

Steiner triple systems

For $k = 3$ and $\lambda = 1$, such a design is a Steiner triple system. The blocks are 3-subsets of the set of points, and two distinct points lie in a unique block.
The two equations for a Steiner triple system assert that

$$2r = v - 1, \qquad 3b = vr,$$

so that $r = (v-1)/2$ and $b = v(v-1)/6$. The condition that these are integers shows that $v \equiv 1$ or $3 \pmod 6$.

Steiner triple systems

For $k = 3$ and $\lambda = 1$, such a design is a Steiner triple system. The blocks are 3-subsets of the set of points, and two distinct points lie in a unique block.

The two equations for a Steiner triple system assert that

$$2r = v - 1, \qquad 3b = vr,$$

so that $r = (v-1)/2$ and $b = v(v-1)/6$. The condition that these are integers shows that $v \equiv 1$ or $3 \pmod 6$.

In the nineteenth century, Thomas Kirkman showed that this necessary condition is also sufficient for the existence of a Steiner triple system.

Wilson's Theorem

In the early 1970s, Wilson discovered a far-reaching generalisation of this theorem. His result has wide applicability; we quote it just for BIBDs.

Wilson's Theorem

In the early 1970s, Wilson discovered a far-reaching generalisation of this theorem. His result has wide applicability; we quote it just for BIBDs.

Suppose that we have a BIBD with given k and λ. Given v, k, λ, the counting equations show that $r = \lambda(v-1)/(k-1)$ and $b = rv/k = \lambda v(v-1)/k(k-1)$. So a necessary condition is that $k-1$ divides $\lambda(v-1)$ and k divides $\lambda v(v-1)$.

Wilson's Theorem

In the early 1970s, Wilson discovered a far-reaching generalisation of this theorem. His result has wide applicability; we quote it just for BIBDs.

Suppose that we have a BIBD with given k and λ. Given v, k, λ, the counting equations show that $r = \lambda(v-1)/(k-1)$ and $b = rv/k = \lambda v(v-1)/k(k-1)$. So a necessary condition is that $k-1$ divides $\lambda(v-1)$ and k divides $\lambda v(v-1)$.

Theorem

If v is sufficiently large (in terms of k and λ), then the above necessary conditions are also sufficient for the existence of a BIBD.

Wilson's Theorem

In the early 1970s, Wilson discovered a far-reaching generalisation of this theorem. His result has wide applicability; we quote it just for BIBDs.
Suppose that we have a BIBD with given k and λ. Given v, k, λ, the counting equations show that $r = \lambda(v-1)/(k-1)$ and $b = rv/k = \lambda v(v-1)/k(k-1)$. So a necessary condition is that $k-1$ divides $\lambda(v-1)$ and k divides $\lambda v(v-1)$.

Theorem

If v is sufficiently large (in terms of k and λ), then the above necessary conditions are also sufficient for the existence of a BIBD.

Of course, this doesn't tell us either how large v has to be, or what to do if the necessary conditions are not satisfied!

Variance-balanced designs

A block design is variance-balanced if its concurrence matrix is a linear combination of I and the all-1 matrix J. Such a design, if binary, is a BIBD, and hence optimal on all criteria we have discussed; but here we do not assume that the design is binary. For short we write $\mathrm{VB}(v, k, \lambda)$ for a variance-balanced design with given values of these parameters, where λ is the common off-diagonal entry of the concurrence matrix.

Variance-balanced designs

A block design is variance-balanced if its concurrence matrix is a linear combination of I and the all-1 matrix J. Such a design, if binary, is a BIBD, and hence optimal on all criteria we have discussed; but here we do not assume that the design is binary. For short we write VB(v, k, λ) for a variance-balanced design with given values of these parameters, where λ is the common off-diagonal entry of the concurrence matrix.
The non-binary design with $v = 5, k = 3$, and $b = 7$ given earlier, is variance-balanced with $\lambda = 2$:

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

Variance-balanced designs

A block design is variance-balanced if its concurrence matrix is a linear combination of I and the all-1 matrix J. Such a design, if binary, is a BIBD, and hence optimal on all criteria we have discussed; but here we do not assume that the design is binary.

For short we write $\mathrm{VB}(v, k, \lambda)$ for a variance-balanced design with given values of these parameters, where λ is the common off-diagonal entry of the concurrence matrix.

The non-binary design with $v = 5$, $k = 3$, and $b = 7$ given earlier, is variance-balanced with $\lambda = 2$:

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

Treatments 1 and 2 concur twice in the first block; any other pair lie in two different blocks.

Optimality

Variance-balanced designs are not always optimal. Here are two examples of variance-balanced designs with $v = b = 7$ and $k = 6$:

Optimality

Variance-balanced designs are not always optimal. Here are two examples of variance-balanced designs with $v = b = 7$ and $k = 6$:

- the design whose blocks are all the 6-subsets of the set of points;

Optimality

Variance-balanced designs are not always optimal. Here are two examples of variance-balanced designs with $v = b = 7$ and $k = 6$:

- the design whose blocks are all the 6-subsets of the set of points;
- the design obtained from the Fano plane by doubling each occurrence of a point in a block (so that the first block is the multiset $[1, 1, 2, 2, 3, 3]$.

Optimality

Variance-balanced designs are not always optimal. Here are two examples of variance-balanced designs with $v = b = 7$ and $k = 6$:

- the design whose blocks are all the 6-subsets of the set of points;
- the design obtained from the Fano plane by doubling each occurrence of a point in a block (so that the first block is the multiset $[1, 1, 2, 2, 3, 3]$.

The first design, with $\lambda = 5$, is a BIBD, and hence is optimal by Kiefer's Theorem. The second has $\lambda = 4$.

Two questions about variance-balanced designs

Two things we would like to know about variance-balanced designs are

Two questions about variance-balanced designs

Two things we would like to know about variance-balanced designs are

- Given k and λ, for which values of v do $\mathrm{VB}(v,k,\lambda)$ designs exist, and what are the possible numbers of blocks of such designs?
- When are variance-balanced designs optimal in some sense?

Two questions about variance-balanced designs

Two things we would like to know about variance-balanced designs are

- Given k and λ, for which values of v do $\mathrm{VB}(v,k,\lambda)$ designs exist, and what are the possible numbers of blocks of such designs?
- When are variance-balanced designs optimal in some sense?

Morgan and Srivastav have investigated these designs (which they call "completely symmetric").

VB designs with maximal trace

Morgan and Srivastav define two new parameters of a VB design, as follows:

$$r = \left\lfloor \frac{bk}{v} \right\rfloor, \qquad p = bk - vr,$$

so that $bk = vr + p$ and $0 \leq p \leq v - 1$. Thus, in a BIBD we have $p = 0$. Note that the use of r does not here imply that the design has constant replication!

VB designs with maximal trace

Morgan and Srivastav define two new parameters of a VB design, as follows:

$$r = \left\lfloor \frac{bk}{v} \right\rfloor, \qquad p = bk - vr,$$

so that $bk = vr + p$ and $0 \leq p \leq v - 1$. Thus, in a BIBD we have $p = 0$. Note that the use of r does not here imply that the design has constant replication!
Morgan and Srivastav further say that a VB design has maximum trace if its parameters satisfy the equation $r(k-1) = (v-1)\lambda$.

VB designs with maximal trace

Morgan and Srivastav define two new parameters of a VB design, as follows:

$$r = \left\lfloor \frac{bk}{v} \right\rfloor, \qquad p = bk - vr,$$

so that $bk = vr + p$ and $0 \leq p \leq v - 1$. Thus, in a BIBD we have $p = 0$. Note that the use of r does not here imply that the design has constant replication!
Morgan and Srivastav further say that a VB design has maximum trace if its parameters satisfy the equation $r(k-1) = (v-1)\lambda$.
In our examples above, $r = \lfloor 7 \cdot 6/7 \rfloor = 6$ and $p = 0$. Since $r(k-1)/(v-1) = 6 \cdot 5/6 = 5$, we see that the first design has maximal trace, but the second does not.

The reason for the term "maximal trace" is as follows. Since $bk < v(r+1)$, some treatment occurs at most r times on the bk plots. Each occurrence contributes at most $k-1$ edges to the concurrence graph, so the valency of this vertex is at most $r(k-1)$. But the concurrence graph of a VB design is regular, with valency $(v-1)\lambda$; so we have $(v-1)\lambda \leq r(k-1)$, and the trace of the concurrence matrix (which is $v(v-1)\lambda$) is at most $vr(k-1)$; equality for the trace implies that $(v-1)\lambda = r(k-1)$.

The reason for the term “maximal trace” is as follows. Since $bk < v(r+1)$, some treatment occurs at most r times on the bk plots. Each occurrence contributes at most $k-1$ edges to the concurrence graph, so the valency of this vertex is at most $r(k-1)$. But the concurrence graph of a VB design is regular, with valency $(v-1)\lambda$; so we have $(v-1)\lambda \le r(k-1)$, and the trace of the concurrence matrix (which is $v(v-1)\lambda$) is at most $vr(k-1)$; equality for the trace implies that $(v-1)\lambda = r(k-1)$. The above argument shows that, in a VB design of maximum trace, any point lies in at least r blocks (counted with multiplicity), with equality if and only if the point occurs at most once in each block. Since $bk = vr + p$, it follows that the number of “bad” points (which occur more than once in some block) is at most p. So if $p = 0$, the design is binary, and is a BIBD or 2-design.

The reason for the term "maximal trace" is as follows. Since $bk < v(r+1)$, some treatment occurs at most r times on the bk plots. Each occurrence contributes at most $k-1$ edges to the concurrence graph, so the valency of this vertex is at most $r(k-1)$. But the concurrence graph of a VB design is regular, with valency $(v-1)\lambda$; so we have $(v-1)\lambda \leq r(k-1)$, and the trace of the concurrence matrix (which is $v(v-1)\lambda$) is at most $vr(k-1)$; equality for the trace implies that $(v-1)\lambda = r(k-1)$.
The above argument shows that, in a VB design of maximum trace, any point lies in at least r blocks (counted with multiplicity), with equality if and only if the point occurs at most once in each block. Since $bk = vr + p$, it follows that the number of "bad" points (which occur more than once in some block) is at most p. So if $p = 0$, the design is binary, and is a BIBD or 2-design.
In the examples, we have $r = 6$, $p = 0$, confirming that the first design has maximum trace but the second does not.

Optimality

Theorem

A variance-balanced design with maximal trace is E-optimal.

Optimality

Theorem

A variance-balanced design with maximal trace is E-optimal.

This was proved by Morgan and Srivastav.

Optimality

Theorem
A variance-balanced design with maximal trace is E-optimal.

This was proved by Morgan and Srivastav.

Theorem
A variance-balanced design is E-optimal if $k < v$ and the number of non-binary blocks is at most $v/2$.

Optimality

Theorem

A variance-balanced design with maximal trace is E-optimal.

This was proved by Morgan and Srivastav.

Theorem

A variance-balanced design is E-optimal if $k < v$ and the number of non-binary blocks is at most $v/2$.

Proof coming up …

Optimality

Theorem
A variance-balanced design with maximal trace is E-optimal.

This was proved by Morgan and Srivastav.

Theorem
A variance-balanced design is E-optimal if $k < v$ and the number of non-binary blocks is at most $v/2$.

Proof coming up …
It follows that our example of a non-binary design, with $v = 5$, $k = 3$ (which is variance-balanced and has one non-binary block) is E-optimal.

Let x be the number of non-binary blocks. A binary block of size k contributes $k(k-1)/2$ edges to the concurrence graph, while a non-binary block contributes fewer than this number.

Let x be the number of non-binary blocks. A binary block of size k contributes $k(k-1)/2$ edges to the concurrence graph, while a non-binary block contributes fewer than this number. So the number of edges (which we know to be $\lambda v(v-1)/2$) is at most $bk(k-1)/2 - x$, so that $x \geq (bk(k-1) - \lambda v(v-1))/2$. This gives

$$b \leq \frac{\lambda v(v-1) + 2x}{k(k-1)}.$$

Let x be the number of non-binary blocks. A binary block of size k contributes $k(k-1)/2$ edges to the concurrence graph, while a non-binary block contributes fewer than this number. So the number of edges (which we know to be $\lambda v(v-1)/2$) is at most $bk(k-1)/2 - x$, so that $x \geq (bk(k-1) - \lambda v(v-1))/2$. This gives

$$b \leq \frac{\lambda v(v-1) + 2x}{k(k-1)}.$$

The non-trivial Laplacian eigenvalues of the λ-fold complete graph are all equal to λv. So, if our design is not E-optimal, then a E-better design (with the same values of (v, b, k)) has least Laplacian eigenvalue greater than λv.

Let δ be the minimal degree of the concurrence graph of such a design. Then δ edges separate a single vertex from the rest of the graph.

Let δ be the minimal degree of the concurrence graph of such a design. Then δ edges separate a single vertex from the rest of the graph.
By the Cutset Lemma,

$$\lambda v < \mu_1 \leq \delta(1 + 1/(v-1)) = \delta v/(v-1),$$

so that $\delta > \lambda(v-1)$, or $\delta \geq \lambda(v-1) + 1$.

Let δ be the minimal degree of the concurrence graph of such a design. Then δ edges separate a single vertex from the rest of the graph.
By the Cutset Lemma,

$$\lambda v < \mu_1 \leq \delta(1 + 1/(v-1)) = \delta v/(v-1),$$

so that $\delta > \lambda(v-1)$, or $\delta \geq \lambda(v-1) + 1$.
Hence the concurrence graph has at least $v(\lambda(v-1)+1)/2$ edges. Since each block of this design contributes at most $k(k-1)/2$ edges, we have

$$b \geq \frac{v(\lambda(v-1)+1)}{k(k-1)}.$$

Let δ be the minimal degree of the concurrence graph of such a design. Then δ edges separate a single vertex from the rest of the graph.
By the Cutset Lemma,

$$\lambda v < \mu_1 \leq \delta(1 + 1/(v-1)) = \delta v/(v-1),$$

so that $\delta > \lambda(v-1)$, or $\delta \geq \lambda(v-1) + 1$.
Hence the concurrence graph has at least $v(\lambda(v-1)+1)/2$ edges. Since each block of this design contributes at most $k(k-1)/2$ edges, we have

$$b \geq \frac{v(\lambda(v-1)+1)}{k(k-1)}.$$

Combining these two bounds for b, we see that $x \geq v/2$. So, if $x < v/2$, then no E-better design can exist.

Existence of VB designs of maximal trace

If we have two VB designs on the same set of v points with the same block size k, having parameters λ_1 and λ_2, then the multiset union of the block multisets is again VB, with parameter $\lambda_1 + \lambda_2$. The new design is not necessarily of maximum trace; but it is so if one of the VB designs we start with is a BIBD and the other is of maximum trace, or if the sum of their p parameters is less than v.

Existence of VB designs of maximal trace

If we have two VB designs on the same set of v points with the same block size k, having parameters λ_1 and λ_2, then the multiset union of the block multisets is again VB, with parameter $\lambda_1 + \lambda_2$. The new design is not necessarily of maximum trace; but it is so if one of the VB designs we start with is a BIBD and the other is of maximum trace, or if the sum of their p parameters is less than v.

For example, suppose that $k = 3$. A VB design of maximum trace satisfies $2r = (v-1)\lambda$, so that λ is even or v is odd. Moreover, $\lambda = 1$ is impossible (except for Steiner triple systems), since a non-binary block gives concurrence at least 2. Morgan and Srivastav proved that these necessary conditions are sufficient:

Existence of VB designs of maximal trace

If we have two VB designs on the same set of v points with the same block size k, having parameters λ_1 and λ_2, then the multiset union of the block multisets is again VB, with parameter $\lambda_1 + \lambda_2$. The new design is not necessarily of maximum trace; but it is so if one of the VB designs we start with is a BIBD and the other is of maximum trace, or if the sum of their p parameters is less than v.

For example, suppose that $k = 3$. A VB design of maximum trace satisfies $2r = (v-1)\lambda$, so that λ is even or v is odd. Moreover, $\lambda = 1$ is impossible (except for Steiner triple systems), since a non-binary block gives concurrence at least 2. Morgan and Srivastav proved that these necessary conditions are sufficient:

Theorem

A $\mathrm{VB}(v, 3, \lambda)$ *design of maximum trace exists whenever* $\lambda(v-1)$ *is even and* $\lambda > 1$*.*

Proof

A BIBD with $k = 3$ and $\lambda = 6$ exists for all v. So it is enough to settle the existence question for λ in a complete set of non-zero residues mod 6. Now BIBDs exist in the following cases:

Proof

A BIBD with $k = 3$ and $\lambda = 6$ exists for all v. So it is enough to settle the existence question for λ in a complete set of non-zero residues mod 6. Now BIBDs exist in the following cases:

- for $\lambda = 1$ or 5, if $v \equiv 1$ or 3 mod 6;
- for $\lambda = 2$ or 4, if $v \equiv 0$ or 1 mod 3;
- for $\lambda = 3$, if v is odd.

Proof

A BIBD with $k = 3$ and $\lambda = 6$ exists for all v. So it is enough to settle the existence question for λ in a complete set of non-zero residues mod 6. Now BIBDs exist in the following cases:

- for $\lambda = 1$ or 5, if $v \equiv 1$ or 3 mod 6;
- for $\lambda = 2$ or 4, if $v \equiv 0$ or 1 mod 3;
- for $\lambda = 3$, if v is odd.

We construct VB designs for $\lambda = 2$ and $v \equiv 2$ mod 3; they have $p = 1$, so the union of two copies settles $\lambda = 4$. For $\lambda = 5$ or $\lambda = 7$, with v odd, there is a BIBD unless $v \equiv 5$ mod 6; in that case we can take a 2-design with $\lambda = 3$ and a VB design with $\lambda = 2$ or $\lambda = 4$.

Here is a construction for VB$(v, 3, 2)$ designs having just one non-binary block. In this case, as we have seen, we must have $v \equiv 2 \bmod 3$.

Here is a construction for $\mathrm{VB}(v,3,2)$ designs having just one non-binary block. In this case, as we have seen, we must have $v \equiv 2 \bmod 3$.
Suppose first that $v \equiv 2 \bmod 6$. There exist Steiner triple systems of orders $v \pm 1$. Take two such systems, on the point sets $\{1,\ldots,v+1\}$ and $\{1,\ldots,v-1\}$ respectively; let the sets of blocks be $\mathcal{B}_1$ and $\mathcal{B}_2$. Without loss of generality, suppose that the third point of the block B of $\mathcal{B}_1$ containing v and $v+1$ is $v-1$.

Here is a construction for VB$(v,3,2)$ designs having just one non-binary block. In this case, as we have seen, we must have $v \equiv 2 \bmod 3$.

Suppose first that $v \equiv 2 \bmod 6$. There exist Steiner triple systems of orders $v \pm 1$. Take two such systems, on the point sets $\{1,\ldots,v+1\}$ and $\{1,\ldots,v-1\}$ respectively; let the sets of blocks be $\mathcal{B}_1$ and $\mathcal{B}_2$. Without loss of generality, suppose that the third point of the block B of $\mathcal{B}_1$ containing v and $v+1$ is $v-1$.

Now we take the point set of the new design to be $\{1,\ldots,v\}$. For the blocks, we first remove the block B from $\mathcal{B}_1$; then we replace each occurrence of $v+1$ in any other block with v; the resulting blocks together with $[v-1,v-1,v]$ make up the design.

We have to check that $\{v-1, v\}$ lies only in $[v-1, v-1, v]$, while every other pair $\{i, j\}$ lies in two blocks. For the first, note that the only other candidate, namely B, has been removed. For the second, there are two cases:

We have to check that $\{v-1, v\}$ lies only in $[v-1, v-1, v]$, while every other pair $\{i, j\}$ lies in two blocks. For the first, note that the only other candidate, namely B, has been removed. For the second, there are two cases:

- $j = v, i \neq v-1$: in $\mathcal{B}_1$, there is one block containing i and v, and one containing i and $v+1$ (in which $v+1$ is replaced by v). No block of $\mathcal{B}_2$ can occur.

We have to check that $\{v-1, v\}$ lies only in $[v-1, v-1, v]$, while every other pair $\{i, j\}$ lies in two blocks. For the first, note that the only other candidate, namely B, has been removed. For the second, there are two cases:

- $j = v$, $i \neq v-1$: in $\mathcal{B}_1$, there is one block containing i and v, and one containing i and $v+1$ (in which $v+1$ is replaced by v). No block of $\mathcal{B}_2$ can occur.
- $v \notin \{i, j\}$: one block of $\mathcal{B}_1$ and one of $\mathcal{B}_2$ contain $\{i, j\}$, and these two points are unchanged in these blocks.

There is a similar but more elaborate construction when $v \equiv 5 \bmod 6$. In this case, both $v-2$ and $v+2$ are orders of Steiner triple systems.

There is a similar but more elaborate construction when $v \equiv 5 \bmod 6$. In this case, both $v-2$ and $v+2$ are orders of Steiner triple systems.
Since there are many non-isomorphic Steiner triple systems, this construction gives rise to many VB designs with $k=3$.

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

- Six triangles and one double edge ($b = 7$): we saw an example.

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

- Six triangles and one double edge ($b = 7$): we saw an example.
- Four triangles and four double edges ($b = 8$): take the BIBD consisting of all the 3-subsets of a 4-set and join its four points to the fifth point by four double edges.

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

- Six triangles and one double edge ($b = 7$): we saw an example.
- Four triangles and four double edges ($b = 8$): take the BIBD consisting of all the 3-subsets of a 4-set and join its four points to the fifth point by four double edges.
- Two triangles and seven double edges ($b = 9$): take a triangle twice and double the seven uncovered edges.

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

- Six triangles and one double edge ($b = 7$): we saw an example.
- Four triangles and four double edges ($b = 8$): take the BIBD consisting of all the 3-subsets of a 4-set and join its four points to the fifth point by four double edges.
- Two triangles and seven double edges ($b = 9$): take a triangle twice and double the seven uncovered edges.
- Ten double edges ($b = 10$): this is a boring design with all its blocks non-binary.

Example

Consider the case $v = 5$, $k = 3$, $\lambda = 2$. Each block contributes either a triangle or a double edge to the concurrence graph, depending on whether or not it is binary. There are four cases:

- Six triangles and one double edge ($b = 7$): we saw an example.
- Four triangles and four double edges ($b = 8$): take the BIBD consisting of all the 3-subsets of a 4-set and join its four points to the fifth point by four double edges.
- Two triangles and seven double edges ($b = 9$): take a triangle twice and double the seven uncovered edges.
- Ten double edges ($b = 10$): this is a boring design with all its blocks non-binary.

The values of (r, p) in the four cases are $(4, 1)$, $(4, 4)$, $(5, 2)$ and $(6, 0)$. So the first two have maximum trace; the others don't.

Is G a concurrence graph?

Given a graph G on v vertices, and an integer k, we would like to know: *Is G the concurrence graph of a block design with block size k?*

Is G a concurrence graph?

Given a graph G on v vertices, and an integer k, we would like to know: *Is G the concurrence graph of a block design with block size k?*

If $k = 2$, then the graph is "the same" as the design; the blocks are just edges of the graph.

Is G a concurrence graph?

Given a graph G on v vertices, and an integer k, we would like to know: *Is G the concurrence graph of a block design with block size k?*
If $k = 2$, then the graph is "the same" as the design; the blocks are just edges of the graph.
If the design is binary, then each block contributes a complete graph of size k; so we have to decide whether G is the edge-disjoint union of complete graphs of size k. This is the question which is answered by Wilson's theorem in the case of the λ-fold complete graph. In general, it is necessary that every vertex has valency divisible by $k - 1$, and the total number of edges is divisible by $k(k - 1)/2$.

Is G a concurrence graph?

Given a graph G on v vertices, and an integer k, we would like to know: *Is G the concurrence graph of a block design with block size k?*

If $k = 2$, then the graph is "the same" as the design; the blocks are just edges of the graph.

If the design is binary, then each block contributes a complete graph of size k; so we have to decide whether G is the edge-disjoint union of complete graphs of size k. This is the question which is answered by Wilson's theorem in the case of the λ-fold complete graph. In general, it is necessary that every vertex has valency divisible by $k - 1$, and the total number of edges is divisible by $k(k-1)/2$.

What happens in general?

Weighted cliques

Let $w_1, w_2, \ldots, w_m$ be positive integers, where $m > 1$. A weighted clique with weights $w_1, \ldots, w_m$ is a graph on m vertices, in which the ith and jth vertices are joined by $w_i w_j$ edges. Its weight is the sum of the weights w_i.

Weighted cliques

Let $w_1, w_2, \ldots, w_m$ be positive integers, where $m > 1$. A weighted clique with weights $w_1, \ldots, w_m$ is a graph on m vertices, in which the ith and jth vertices are joined by $w_i w_j$ edges. Its weight is the sum of the weights w_i.

If all w_i are equal to 1, this is just a complete graph on m vertices, and has weight m.

Weighted cliques

Let $w_1, w_2, \ldots, w_m$ be positive integers, where $m > 1$. A weighted clique with weights $w_1, \ldots, w_m$ is a graph on m vertices, in which the ith and jth vertices are joined by $w_i w_j$ edges. Its weight is the sum of the weights w_i.

If all w_i are equal to 1, this is just a complete graph on m vertices, and has weight m.

Theorem

The graph G is the concurrence graph of a block design with block size k if and only if it is an edge-disjoint union of weighted cliques each with weight k.

Weighted cliques

Let $w_1, w_2, \ldots, w_m$ be positive integers, where $m > 1$. A weighted clique with weights $w_1, \ldots, w_m$ is a graph on m vertices, in which the ith and jth vertices are joined by $w_i w_j$ edges. Its weight is the sum of the weights w_i.
If all w_i are equal to 1, this is just a complete graph on m vertices, and has weight m.

Theorem

The graph G is the concurrence graph of a block design with block size k if and only if it is an edge-disjoint union of weighted cliques each with weight k.

In the design, if a weighted clique with weights $w_1, \ldots, w_m$ corresponds to block j, then the weights are equal to the incidence matrix entries N_{ij} for appropriate values of i.

Weighted cliques

Let $w_1, w_2, \ldots, w_m$ be positive integers, where $m > 1$. A weighted clique with weights $w_1, \ldots, w_m$ is a graph on m vertices, in which the ith and jth vertices are joined by $w_i w_j$ edges. Its weight is the sum of the weights w_i.

If all w_i are equal to 1, this is just a complete graph on m vertices, and has weight m.

Theorem

The graph G is the concurrence graph of a block design with block size k if and only if it is an edge-disjoint union of weighted cliques each with weight k.

In the design, if a weighted clique with weights $w_1, \ldots, w_m$ corresponds to block j, then the weights are equal to the incidence matrix entries N_{ij} for appropriate values of i.

This generalizes the "graph decomposition" interpretation of BIBDs. As we saw, the weighted cliques of weight 3 are a triangle and a double edge.

Decomposition into weighted cliques

Usually the decomposition into weighted cliques, if it exists, is far from unique.

Decomposition into weighted cliques

Usually the decomposition into weighted cliques, if it exists, is far from unique.

- The Fano plane arises from a decomposition of the 21 edges of K_7 into seven triangles. It is unique up to isomorphism, but there are 30 different ways to make the decomposition (corresponding to the fact that the automorphism group of the Fano plane has index 30 in the symmetric group S_7).

Decomposition into weighted cliques

Usually the decomposition into weighted cliques, if it exists, is far from unique.

- The Fano plane arises from a decomposition of the 21 edges of K_7 into seven triangles. It is unique up to isomorphism, but there are 30 different ways to make the decomposition (corresponding to the fact that the automorphism group of the Fano plane has index 30 in the symmetric group S_7).
- In our variance-balanced design with $v = 5$, $k = 3$ and $b = 7$, we took the block $[1, 1, 2]$. However, the block $[1, 2, 2]$ would have been just as good, and would have given us the same concurrence graph.

Other graph parameters

For the final part of the course, we turn to something completely different …

Other graph parameters

For the final part of the course, we turn to something completely different ...

The number of spanning trees of a graph (the D-optimality parameter) also happens to be an evaluation of a famous two-variable polynomial, the Tutte polynomial, of the graph.

Other evaluations of the Tutte polynomial give lots more information about the graph: number of proper colourings with a given number of colours, number of acyclic or totally cyclic orientations, etc.

Other graph parameters

For the final part of the course, we turn to something completely different . . .

The number of spanning trees of a graph (the D-optimality parameter) also happens to be an evaluation of a famous two-variable polynomial, the Tutte polynomial, of the graph.

Other evaluations of the Tutte polynomial give lots more information about the graph: number of proper colourings with a given number of colours, number of acyclic or totally cyclic orientations, etc.

We will define the Tutte polynomial and consider how it is related to some of the invariants we have met.

The chromatic polynomial

A proper colouring of a graph with q colours is an assignment of colours to the vertices so that adjacent vertices get different colours.

The chromatic polynomial

A proper colouring of a graph with q colours is an assignment of colours to the vertices so that adjacent vertices get different colours.
It is well-known that the number of proper colourings of G is the evaluation at q of a monic polynomial of degree $n = |V(G)|$, known as the chromatic polynomial of G.

The chromatic polynomial

A proper colouring of a graph with q colours is an assignment of colours to the vertices so that adjacent vertices get different colours.

It is well-known that the number of proper colourings of G is the evaluation at q of a monic polynomial of degree $n = |V(G)|$, known as the chromatic polynomial of G.

This is usually proved by "deletion-contraction". It suits my purpose here to give a different proof, using "inclusion-exclusion".

Let S be the the set of all (proper or not) vertex-colourings of G with q colours. For any edge e, let T_e be the set of colourings for which e is improper (has both ends of the same colour); and for $A \subseteq E(G)$ let

$$T_A = \bigcap_{e \in A} T_e,$$

with $T_\emptyset = S$ by convention.

Let S be the the set of all (proper or not) vertex-colourings of G with q colours. For any edge e, let T_e be the set of colourings for which e is improper (has both ends of the same colour); and for $A \subseteq E(G)$ let

$$T_A = \bigcap_{e \in A} T_e,$$

with $T_\emptyset = S$ by convention.

Let $k(A)$ be the number of connected components of the graph $(V(G), A)$. A colouring in $T(A)$ has the property that all the vertices in a connected component of this graph have the same colour. So $|T_A| = q^{k(A)}$. Then there are $q^{k(A)}$ colourings in which all the edges in A are bad.

Let S be the the set of all (proper or not) vertex-colourings of G with q colours. For any edge e, let T_e be the set of colourings for which e is improper (has both ends of the same colour); and for $A \subseteq E(G)$ let

$$T_A = \bigcap_{e \in A} T_e,$$

with $T_\emptyset = S$ by convention.
Let $k(A)$ be the number of connected components of the graph $(V(G), A)$. A colouring in $T(A)$ has the property that all the vertices in a connected component of this graph have the same colour. So $|T_A| = q^{k(A)}$. Then there are $q^{k(A)}$ colourings in which all the edges in A are bad.
So by PIE, the number of proper colourings is

$$\sum_{A \subseteq E(G)} (-1)^{|A|} q^{k(A)} = P_G(q),$$

where P_G is the chromatic polynomial of G.

Rank

The rank $r(A)$ of a set A of edges of a graph G on n vertices is defined to be the cardinality of the largest acyclic subset of A. It is easy to see that this is $n - k(A)$.

Rank

The rank $r(A)$ of a set A of edges of a graph G on n vertices is defined to be the cardinality of the largest acyclic subset of A. It is easy to see that this is $n - k(A)$.

Rank has another interpretation. Recall the (signed) vertex-edge incidence matrix Q of G, as defined in Lecture 2. Then $r(A)$ is the rank (in the sense of linear algebra) of the submatrix formed by the columns indexed by edges in A. The proof is an exercise.

Rank

The rank $r(A)$ of a set A of edges of a graph G on n vertices is defined to be the cardinality of the largest acyclic subset of A. It is easy to see that this is $n - k(A)$.

Rank has another interpretation. Recall the (signed) vertex-edge incidence matrix Q of G, as defined in Lecture 2. Then $r(A)$ is the rank (in the sense of linear algebra) of the submatrix formed by the columns indexed by edges in A. The proof is an exercise.

In particular, if $G = (V, E)$ is connected, then $r(E) = n - 1$.

The Tutte polynomial

The Tutte polynomial of the graph $G = (V, E)$ is the polynomial

$$T_G(x, y) = \sum_{A \subseteq E} (x-1)^{r(E)-r(A)} (y-1)^{|A|-r(A)}.$$

The Tutte polynomial

The Tutte polynomial of the graph $G = (V, E)$ is the polynomial

$$T_G(x,y) = \sum_{A \subseteq E} (x-1)^{r(E)-r(A)} (y-1)^{|A|-r(A)}.$$

Many important graph parameters are obtained by plugging in special values for x and y, possibly multiplying by a simple factor.

The Tutte polynomial

The Tutte polynomial of the graph $G = (V, E)$ is the polynomial

$$T_G(x,y) = \sum_{A \subseteq E} (x-1)^{r(E)-r(A)} (y-1)^{|A|-r(A)}.$$

Many important graph parameters are obtained by plugging in special values for x and y, possibly multiplying by a simple factor.
In particular, putting $x = y = 2$, every term is 1, so that $T_G(2,2) = 2^{|E|}$.

Other specialisations

Assume that G is connected.

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.
Hence

- $T_G(1, 2)$ is the number of connected spanning subgraphs of G;

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.
Hence

- $T_G(1, 2)$ is the number of connected spanning subgraphs of G;
- $T_G(2, 1)$ is the number of spanning forests of G;

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.
Hence

- $T_G(1, 2)$ is the number of connected spanning subgraphs of G;
- $T_G(2, 1)$ is the number of spanning forests of G;
- $T_G(1, 1)$ is the number of spanning trees of G.

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.
Hence

- $T_G(1, 2)$ is the number of connected spanning subgraphs of G;
- $T_G(2, 1)$ is the number of spanning forests of G;
- $T_G(1, 1)$ is the number of spanning trees of G.

Note that $T_G(1, 1)$ is the number associated with D-optimality!

Other specialisations

Assume that G is connected.
Putting $x = 1$, the only non-zero terms are those which have the exponent of $(x - 1)$ equal to 0, that is, $r(A) = r(E)$, so that the graph (V, A) is connected. Similarly, putting $y = 1$, the only non-zero terms are those with $|A| = r(A)$, in other words, the set A contains no cycles.
Hence

- $T_G(1, 2)$ is the number of connected spanning subgraphs of G;
- $T_G(2, 1)$ is the number of spanning forests of G;
- $T_G(1, 1)$ is the number of spanning trees of G.

Note that $T_G(1, 1)$ is the number associated with D-optimality!
However, other optimality parameters don't appear to be specialisations of the Tutte polynomial.

Examples

Neither of the Tutte polynomial and the Laplacian spectrum of G determines the other.

Examples

Neither of the Tutte polynomial and the Laplacian spectrum of G determines the other.
In one direction, all trees on n vertices have Tutte polynomial x^{n-1}; but we have seen that they can be very different on A- or E-optimality, and hence on Laplacian spectra.

Examples

Neither of the Tutte polynomial and the Laplacian spectrum of G determines the other.
In one direction, all trees on n vertices have Tutte polynomial x^{n-1}; but we have seen that they can be very different on A- or E-optimality, and hence on Laplacian spectra.
In the other direction, the two strongly regular graphs with the same parameters on 16 vertices have the same Laplacian spectra, but have different Tutte polynomials: see below.

Chromatic polynomial revisited

The formula for the Tutte polynomial looks very similar to the formula we deduced for the chromatic polynomial. Indeed, a little persistence shows that, for a connected graph G,

$$P_G(q) = (-1)^{n-1} q T_G(0, -q+1),$$

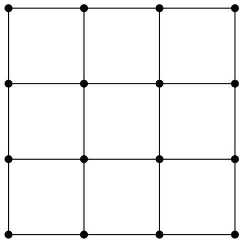
so the numbers of colourings are values of T_G at integer points on the negative real axis.

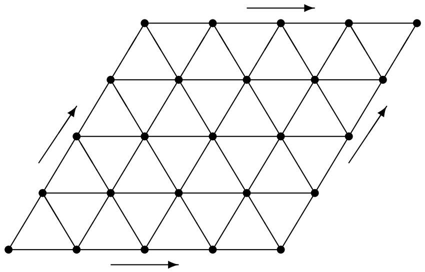
Two strongly regular graphs

Consider the following pair (G_1, G_2) of graphs: on the left, the 4×4 square lattice graph (in which vertices in the same row or column are joined), and on the right, the Shrikhande graph (which is shown drawn on a torus: nearest neighbours are joined, and opposite edges are identified).

Two strongly regular graphs

Consider the following pair (G_1, G_2) of graphs: on the left, the 4×4 square lattice graph (in which vertices in the same row or column are joined), and on the right, the Shrikhande graph (which is shown drawn on a torus: nearest neighbours are joined, and opposite edges are identified).





Each graph is strongly regular with parameters $(16, 6, 2, 2)$: there are 16 vertices, each vertex has valency 6, and any two vertices have 2 common neighbours, whether or not they are joined.

Each graph is strongly regular with parameters $(16, 6, 2, 2)$: there are 16 vertices, each vertex has valency 6, and any two vertices have 2 common neighbours, whether or not they are joined.
So the adjacency matrices satisfy $A^2 = 4I + 2J$, and have eigenvalues 6, 2 and -2; the Laplacian eigenvalues are 0, 4 and 8.

Each graph is strongly regular with parameters $(16, 6, 2, 2)$: there are 16 vertices, each vertex has valency 6, and any two vertices have 2 common neighbours, whether or not they are joined.

So the adjacency matrices satisfy $A^2 = 4I + 2J$, and have eigenvalues 6, 2 and -2; the Laplacian eigenvalues are 0, 4 and 8.

Each graph is associated with a lattice design. Take a Latin square of order 4, and form a graph whose vertices are the cells, two vertices adjacent if they are *not* in the same row or column or contain the same symbol.

Each graph is strongly regular with parameters $(16, 6, 2, 2)$: there are 16 vertices, each vertex has valency 6, and any two vertices have 2 common neighbours, whether or not they are joined.

So the adjacency matrices satisfy $A^2 = 4I + 2J$, and have eigenvalues 6, 2 and -2; the Laplacian eigenvalues are 0, 4 and 8.

Each graph is associated with a lattice design. Take a Latin square of order 4, and form a graph whose vertices are the cells, two vertices adjacent if they are *not* in the same row or column or contain the same symbol.

There are two essentially different Latin squares of order 4: they are the Cayley tables of the Klein group V_4 and the cyclic group C_4. The corresponding graphs are the lattice graph $L_2(4)$ and the Shrikhande graph respectively.

Colouring the graphs

A colouring of the square lattice with four colours is nothing but a Latin square of order 4 (see below). There are 576 Latin squares, and hence $P_{G_1}(4) = 576$.

Colouring the graphs

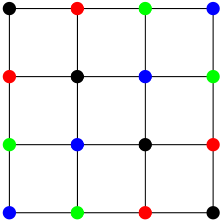
A colouring of the square lattice with four colours is nothing but a Latin square of order 4 (see below). There are 576 Latin squares, and hence $P_{G_1}(4) = 576$.

However, calculation shows that $P_{G_2}(4) = 240$; so the chromatic polynomials, and hence the Tutte polynomials, are different.

Colouring the graphs

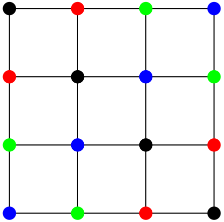
A colouring of the square lattice with four colours is nothing but a Latin square of order 4 (see below). There are 576 Latin squares, and hence $P_{G_1}(4) = 576$.

However, calculation shows that $P_{G_2}(4) = 240$; so the chromatic polynomials, and hence the Tutte polynomials, are different.



Colouring the graphs

A colouring of the square lattice with four colours is nothing but a Latin square of order 4 (see below). There are 576 Latin squares, and hence $P_{G_1}(4) = 576$.
However, calculation shows that $P_{G_2}(4) = 240$; so the chromatic polynomials, and hence the Tutte polynomials, are different.



Each colour gives the positions of a symbol in a Latin square.

Orientations

An orientation of the edges of a graph G is acyclic if there are no directed cycles; it is totally cyclic if every edge is contained in a directed cycle.
Richard Stanley showed that the number $a(G)$ of acyclic orientations of G is

$$a(G) = |P_G(-1)| = |T_G(0, 2)|.$$

It is also known that the number of totally cyclic orientations is $c(G) = |T_G(2, 0)|$.

Orientations

An orientation of the edges of a graph G is acyclic if there are no directed cycles; it is totally cyclic if every edge is contained in a directed cycle.
Richard Stanley showed that the number $a(G)$ of acyclic orientations of G is

$$a(G) = |P_G(-1)| = |T_G(0, 2)|.$$

It is also known that the number of totally cyclic orientations is $c(G) = |T_G(2, 0)|$.
Now recall that the number of spanning trees is $t(G) = T_G(1, 1)$.

The Merino–Welsh conjecture

These three numbers are connected by a remarkable conjecture of Merino and Welsh:

Conjecture

If G has no loops or bridges, then $t(G) \leq \max\{a(G), c(G)\}$.

That is, either the number of acyclic orientations or the number of totally cyclic orientations dominates the number of spanning trees.

The Merino–Welsh conjecture

These three numbers are connected by a remarkable conjecture of Merino and Welsh:

Conjecture

If G has no loops or bridges, then $t(G) \leq \max\{a(G), c(G)\}$.

That is, either the number of acyclic orientations or the number of totally cyclic orientations dominates the number of spanning trees.
The best result so far is by Carsten Thomassen, who showed that this is true for sufficiently sparse graphs (where the number of acyclic orientations wins) and for sufficiently dense graphs (where the number of totally cyclic orientations wins).

Thomassen's Theorem

Theorem

Let G be a connected graph without loops or bridges.

- *If G has at least $4n$ edges, then $t(G) \leq c(G)$.*

Thomassen's Theorem

Theorem

Let G be a connected graph without loops or bridges.

- *If G has at least $4n$ edges, then $t(G) \leq c(G)$.*
- *If G has at most $16n/15$ edges, then $t(G) \leq a(G)$.*

Thomassen's Theorem

Theorem

Let G be a connected graph without loops or bridges.

- *If G has at least* $4n$ *edges, then* $t(G) \leq c(G)$.
- *If G has at most* $16n/15$ *edges, then* $t(G) \leq a(G)$.

The first result applies if the average valency is at least 8; the second if it is at most $32/15 = 2.133\ldots$.

Jackson's Theorem

A related theorem of Bill Jackson is intriguing:

Theorem

Let G be a connected graph without loops or bridges. Then

$$T_G(1,1)^2 \leq T_G(0,3) \cdot T_G(3,0).$$

Jackson's Theorem

A related theorem of Bill Jackson is intriguing:

Theorem

Let G be a connected graph without loops or bridges. Then

$$T_G(1,1)^2 \leq T_G(0,3) \cdot T_G(3,0).$$

Of course, replacing 3 by 2 would give a strengthening of the Merino–Welsh conjecture!

Open problem: a tipping point?

We end with some research problems.

Open problem: a tipping point?

We end with some research problems.

Problem

Given k, is there a number r_0 such that, for designs with block size k and average replication greater than r_0, the different optimality conditions agree?

Open problem: a tipping point?

We end with some research problems.

Problem

Given k, is there a number r_0 such that, for designs with block size k and average replication greater than r_0, the different optimality conditions agree?

The number r_0 might also depend on v. However, work by Robert Johnson and Mark Walters suggests that, for $k = 2$, r_0 might be about 4. This is suggestively similar to Carsten Thomassen's result on the Merino–Welsh conjecture.

Open problem: a tipping point?

We end with some research problems.

Problem

Given k, is there a number r_0 such that, for designs with block size k and average replication greater than r_0, the different optimality conditions agree?

The number r_0 might also depend on v. However, work by Robert Johnson and Mark Walters suggests that, for $k = 2$, r_0 might be about 4. This is suggestively similar to Carsten Thomassen's result on the Merino–Welsh conjecture.

If so, what happens for average replication below r_0? There may be further "phase changes".

Open problem: dense simple graphs

For dense simple graphs (those obtained by removing just a few edges from the complete graph), independent studies by Aylin Cakiroglu and Robert Schumacher suggest that, both for optimality and for maximizing the number of acyclic orientations, the best graphs resemble Turán graphs: that is, the edges removed should be as close as possible to a disjoint union of complete graphs of the same size.

Open problem: dense simple graphs

For dense simple graphs (those obtained by removing just a few edges from the complete graph), independent studies by Aylin Cakiroglu and Robert Schumacher suggest that, both for optimality and for maximizing the number of acyclic orientations, the best graphs resemble Turán graphs: that is, the edges removed should be as close as possible to a disjoint union of complete graphs of the same size.
If we remove complete graphs of the same size, we get a group-divisible design.

Open problem: dense simple graphs

For dense simple graphs (those obtained by removing just a few edges from the complete graph), independent studies by Aylin Cakiroglu and Robert Schumacher suggest that, both for optimality and for maximizing the number of acyclic orientations, the best graphs resemble Turán graphs: that is, the edges removed should be as close as possible to a disjoint union of complete graphs of the same size.
If we remove complete graphs of the same size, we get a group-divisible design.

Problem

Prove the above assertion.

Open problem: adding complete graphs

Aylin Cakiroglu and J. P. Morgan have investigated the following problem. Choose an optimality parameter. For a non-negative integer s, and given v, order the simple graphs on v vertices with a fixed number of edges (or the regular simple graphs of prescribed valency) by the rule that $G_1 <_s G_2$ if the union of G_2 with s copies of K_v beats the union of G_1 with s copies of K_v.

Open problem: adding complete graphs

Aylin Cakiroglu and J. P. Morgan have investigated the following problem. Choose an optimality parameter. For a non-negative integer s, and given v, order the simple graphs on v vertices with a fixed number of edges (or the regular simple graphs of prescribed valency) by the rule that $G_1 <_s G_2$ if the union of G_2 with s copies of K_v beats the union of G_1 with s copies of K_v.

They showed that this order stabilises for sufficiently large s. But in cases which could be computed, it stabilises for $s = 1$ (or at worst for $s = 2$).

Open problem: adding complete graphs

Aylin Cakiroglu and J. P. Morgan have investigated the following problem. Choose an optimality parameter. For a non-negative integer s, and given v, order the simple graphs on v vertices with a fixed number of edges (or the regular simple graphs of prescribed valency) by the rule that $G_1 <_s G_2$ if the union of G_2 with s copies of K_v beats the union of G_1 with s copies of K_v.

They showed that this order stabilises for sufficiently large s. But in cases which could be computed, it stabilises for $s = 1$ (or at worst for $s = 2$).

Problem

Bound the value of s for which the order stabilises in terms of v.

Open problem: adding complete graphs

Aylin Cakiroglu and J. P. Morgan have investigated the following problem. Choose an optimality parameter. For a non-negative integer s, and given v, order the simple graphs on v vertices with a fixed number of edges (or the regular simple graphs of prescribed valency) by the rule that $G_1 <_s G_2$ if the union of G_2 with s copies of K_v beats the union of G_1 with s copies of K_v.

They showed that this order stabilises for sufficiently large s. But in cases which could be computed, it stabilises for $s = 1$ (or at worst for $s = 2$).

Problem

Bound the value of s for which the order stabilises in terms of v.

One can also make the problem "continuous" by expressing the parameter in terms of s and then allowing s to take real values.

Open problem: variance-balanced designs

Problem

Given k and λ, find necessary and sufficient conditions on v for the existence of a variance-balanced design of maximum trace with these values of v, k, λ.

Open problem: variance-balanced designs

Problem

Given k and λ, find necessary and sufficient conditions on v for the existence of a variance-balanced design of maximum trace with these values of v, k, λ.

We saw that this was solved for $k = 3$ and all λ by Morgan and Srivastav.

Open problem: variance-balanced designs

Problem

Given k and λ, find necessary and sufficient conditions on v for the existence of a variance-balanced design of maximum trace with these values of v, k, λ.

We saw that this was solved for $k = 3$ and all λ by Morgan and Srivastav.

More generally, there are theorems about decomposing the edge set of a graph into complete graphs of given sizes; find theorems about decomposing the edge set of a graph into weighted k-cliques, with perhaps some restrictions on the cliques (e.g. as few as possible where the weights are not all 1).

Open problems: Finite geometry

In finite geometry one meets many beautiful and symmetrical block designs of various kinds: generalized polygons, near-polygons, Grassmann geometries, ...

Open problems: Finite geometry

In finite geometry one meets many beautiful and symmetrical block designs of various kinds: generalized polygons, near-polygons, Grassmann geometries, …

Problem

Are these geometries optimal?

Open problems: Finite geometry

In finite geometry one meets many beautiful and symmetrical block designs of various kinds: generalized polygons, near-polygons, Grassmann geometries, . . .

Problem

Are these geometries optimal?

Also one meets geometries of higher rank, that is, with more than two kinds of objects; they will have various rank 2 geometries as truncations and as residuals. These may be relevant in experimental design, if there are several different kinds of treatment, or of "nuisance factor" to be controlled.

Open problems: Finite geometry

In finite geometry one meets many beautiful and symmetrical block designs of various kinds: generalized polygons, near-polygons, Grassmann geometries, …

Problem

Are these geometries optimal?

Also one meets geometries of higher rank, that is, with more than two kinds of objects; they will have various rank 2 geometries as truncations and as residuals. These may be relevant in experimental design, if there are several different kinds of treatment, or of "nuisance factor" to be controlled.

Problem

What is the relation, if any, between optimality of different truncations or residuals of the same higher-rank geometry?

The end

That’s all; thank you all for lasting until the end of the course.

The end

That's all; thank you all for lasting until the end of the course. We may put further information on the course web page at some point. If you are interested in this, or in working on some of these problems, let us know!

The end

That's all; thank you all for lasting until the end of the course. We may put further information on the course web page at some point. If you are interested in this, or in working on some of these problems, let us know!