Antichains and the Structure of Permutation Classes

Robert Brignall

University of Bristol, UK

Monday 8th February, 2010

Outline

Outline

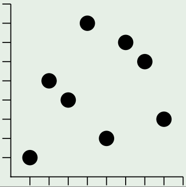
Setting the Scene

- Permutation of length n: an ordering on the symbols $1, \ldots, n$.
- For example: $\pi = 15482763$.

Setting the Scene

- Permutation of length n: an ordering on the symbols $1, \ldots, n$.
- For example: $\pi = 15482763$.
- Graphical viewpoint: plot the points $(i, \pi(i))$.

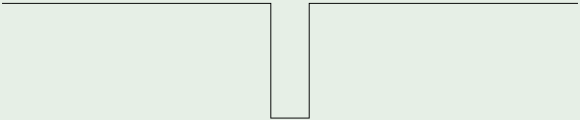
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

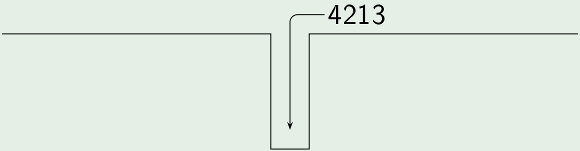
Example



Stack Sorting

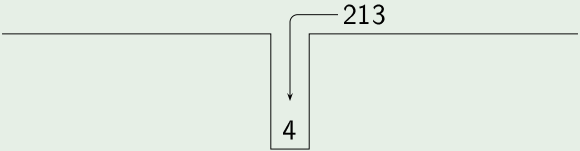
- Knuth (1969): what permutations can be sorted through a stack?

Example



- Knuth (1969): what permutations can be sorted through a stack?

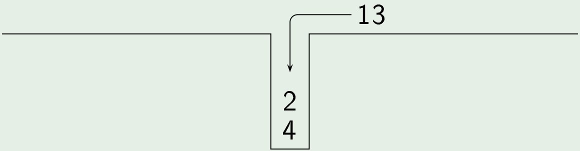
Example



Stack Sorting

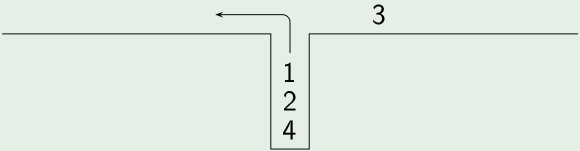
- Knuth (1969): what permutations can be sorted through a stack?

Example



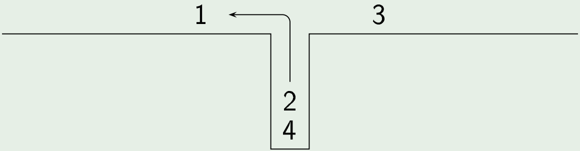
- Knuth (1969): what permutations can be sorted through a stack?

Example



- Knuth (1969): what permutations can be sorted through a stack?

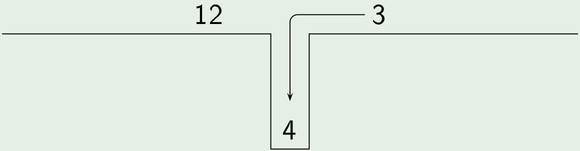
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

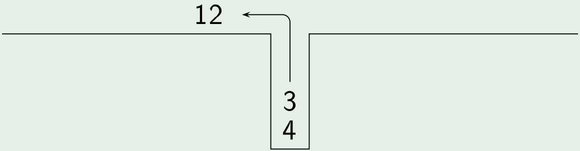
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

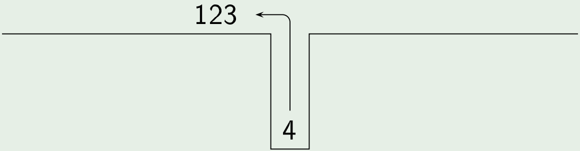
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

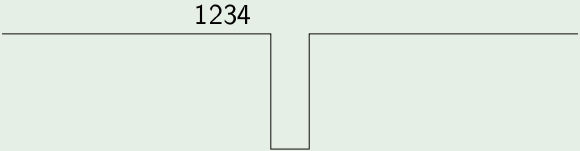
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

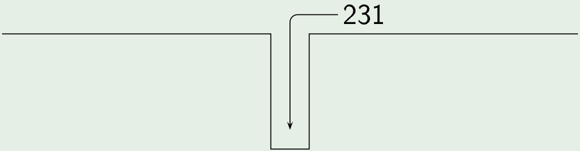
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

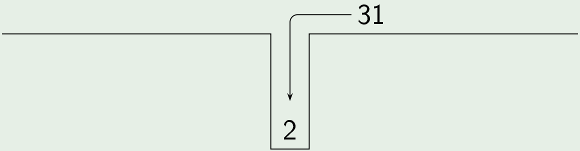
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

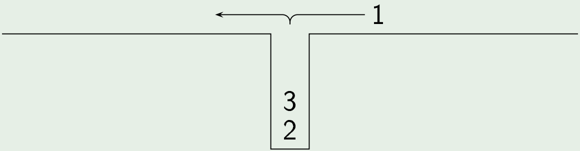
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

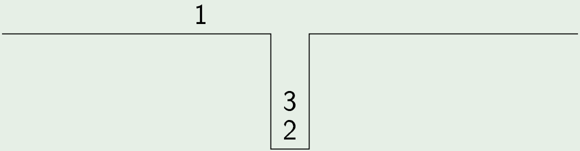
Example



Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

Example

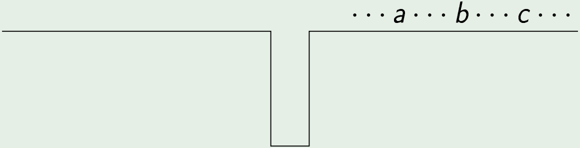


- 231 is not stack-sortable.

Stack Sorting

- Knuth (1969): what permutations can be sorted through a stack?

Example



- 231 is not stack-sortable.
- In general: can't sort any permutation with a subsequence *abc* such that $c < a < b$. (*abc* forms a 231 "pattern".)

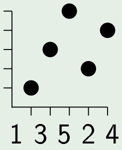
Containment

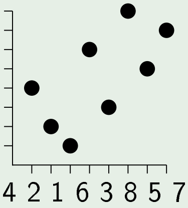
- A permutation $\tau = \tau(1) \cdots \tau(k)$ is contained in the permutation $\sigma = \sigma(1)\sigma(2) \cdots \sigma(n)$ if there exists a subsequence $\sigma(i_1)\sigma(i_2) \cdots \sigma(i_k)$ order isomorphic to τ.

Containment

- A permutation $\tau = \tau(1) \cdots \tau(k)$ is contained in the permutation $\sigma = \sigma(1)\sigma(2) \cdots \sigma(n)$ if there exists a subsequence $\sigma(i_1)\sigma(i_2) \cdots \sigma(i_k)$ order isomorphic to τ.

Example

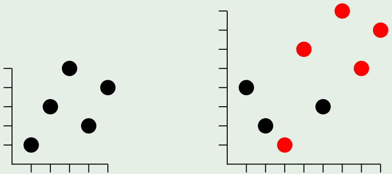




Containment

- A permutation $\tau = \tau(1) \cdots \tau(k)$ is contained in the permutation $\sigma = \sigma(1)\sigma(2) \cdots \sigma(n)$ if there exists a subsequence $\sigma(i_1)\sigma(i_2) \cdots \sigma(i_k)$ order isomorphic to τ.

Example



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

Permutation Classes

- Containment forms a partial order on the set of all permutations.

Permutation Classes

- Containment forms a partial order on the set of all permutations.
- Downwards-closed sets in this partial order form permutation classes. i.e. $\pi \in \mathcal{C}$ and $\sigma \leq \pi$ implies $\sigma \in \mathcal{C}$.

Permutation Classes

- Containment forms a partial order on the set of all permutations.
- Downwards-closed sets in this partial order form permutation classes. i.e. $\pi \in \mathcal{C}$ and $\sigma \leq \pi$ implies $\sigma \in \mathcal{C}$.
- A permutation class $\mathcal{C}$ can be seen to avoid certain permutations. Write $\mathcal{C} = \mathrm{Av}(B) = \{\pi : \beta \not\leq \pi \text{ for all } \beta \in B\}$.

Permutation Classes

- Containment forms a partial order on the set of all permutations.
- Downwards-closed sets in this partial order form permutation classes. i.e. $\pi \in \mathcal{C}$ and $\sigma \leq \pi$ implies $\sigma \in \mathcal{C}$.
- A permutation class $\mathcal{C}$ can be seen to avoid certain permutations. Write $\mathcal{C} = \text{Av}(B) = \{\pi : \beta \not\leq \pi \text{ for all } \beta \in B\}$.
- The minimal avoidance set is the basis. It is unique but need not be finite.
- E.g. the stack-sortable permutations are Av(231).

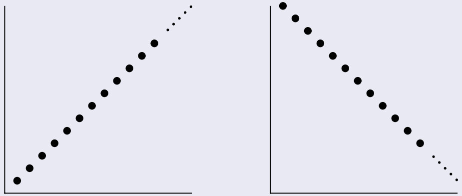
Permutation Classes

- Containment forms a partial order on the set of all permutations.
- Downwards-closed sets in this partial order form permutation classes. i.e. $\pi \in \mathcal{C}$ and $\sigma \leq \pi$ implies $\sigma \in \mathcal{C}$.
- A permutation class $\mathcal{C}$ can be seen to avoid certain permutations. Write $\mathcal{C} = \mathrm{Av}(B) = \{\pi : \beta \not\leq \pi \text{ for all } \beta \in B\}$.
- The minimal avoidance set is the basis. It is unique but need not be finite.
- E.g. the stack-sortable permutations are Av(231).
- Graph theoretic analogue: hereditary properties of graphs (e.g. triangle-free graphs, planar graphs, ...).

Easy Examples

- $\text{Av}(21) = \{1, 12, 123, 1234, \ldots\}$, the increasing permutations.
- $\text{Av}(12) = \{1, 21, 321, 4321, \ldots\}$, the decreasing permutations.

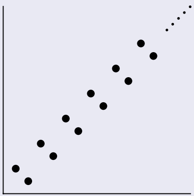
Typical Elements

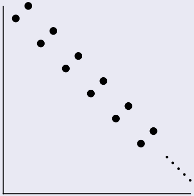


Easy Examples

- $\oplus 21 = \text{Av}(321, 312, 231) = \{1, 12, 21, 123, 132, 213, \ldots\}$.
- $\ominus 12 = \text{Av}(123, 213, 132) = \{1, 12, 21, 231, 312, 321, \ldots\}$.

Typical Elements





Exact Enumeration

- $\mathcal{C}_n$ – permutations in $\mathcal{C}$ of length n.
- $\sum |\mathcal{C}_n| x^n$ is the generating function.

Example

The generating function of $\mathcal{C} = \mathrm{Av}(12)$ is:

$$1 + x + x^2 + x^3 + \cdots \quad = \quad \frac{1}{1 - x}$$

Asymptotic Enumeration

Theorem (Marcus and Tardos, 2004)

For every permutation class $\mathcal{C}$ other than the class of all permutations, there exists a constant K such that

$$\limsup_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|} \leq K.$$

- Upper growth rate of $\mathcal{C}$ is $\limsup_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$.

Asymptotic Enumeration

Theorem (Marcus and Tardos, 2004)

For every permutation class $\mathcal{C}$ other than the class of all permutations, there exists a constant K such that

$$\limsup_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|} \leq K.$$

- Upper growth rate of $\mathcal{C}$ is $\limsup_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$.
- Big open question: does the growth rate, $\lim_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$, always exist?

Av(321) vs Av(231)

- Stack sortable permutations Av(231) enumerated by the Catalan numbers. Generating function:

$$f(x) = \frac{1 - \sqrt{1-4x}}{2x} = 1 + x + 2x^2 + 5x^3 + 14x^4 + \dots$$

Av(321) vs Av(231)

- Stack sortable permutations Av(231) enumerated by the Catalan numbers. Generating function:

$$f(x) = \frac{1-\sqrt{1-4x}}{2x} = 1 + x + 2x^2 + 5x^3 + 14x^4 + \dots$$

- Using the Robinson-Schensted-Knuth correspondence with Young Tableaux, $|\mathrm{Av}(321)|_n = |\mathrm{Av}(231)|_n$.

Av(321) vs Av(231)

- Stack sortable permutations Av(231) enumerated by the Catalan numbers. Generating function:

$$f(x) = \frac{1 - \sqrt{1-4x}}{2x} = 1 + x + 2x^2 + 5x^3 + 14x^4 + \ldots$$

- Using the Robinson-Schensted-Knuth correspondence with Young Tableaux, $|\mathrm{Av}(321)|_n = |\mathrm{Av}(231)|_n$.
- Despite being equinumerous, these two classes are very different: Av(321) contains infinite antichains and hence has uncountably many subclasses, while Av(231) does not.

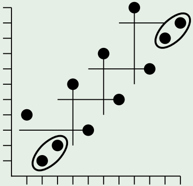
Infinite Antichains

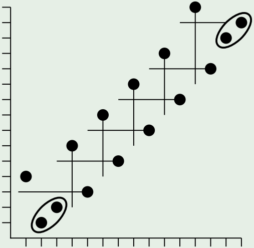
- (Infinite) set of pairwise incomparable permutations.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)



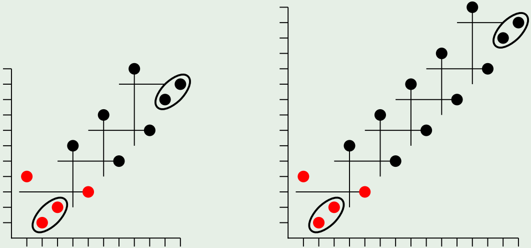


- N.B. These permutations avoid 321.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

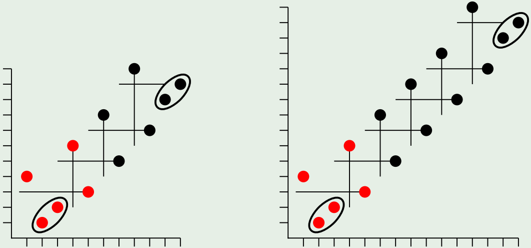


- Bottom copies of 4123 must match up: the anchor.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

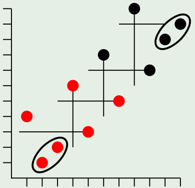


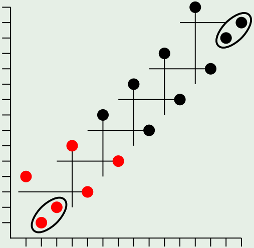
- Each point is matched in turn.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)



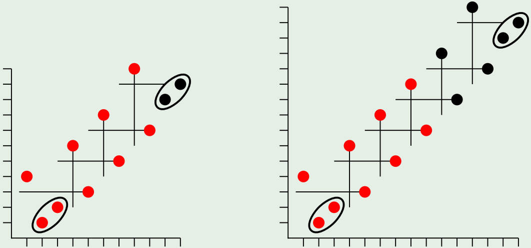


- Each point is matched in turn.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

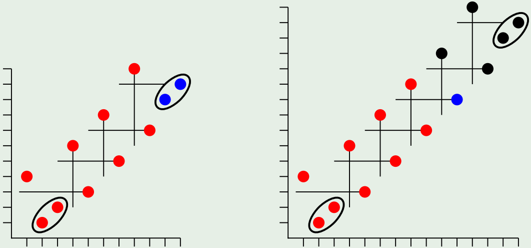


- Each point is matched in turn.

Infinite Antichains

- (Infinite) set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)



- Last pair cannot be embedded.

When are there antichains?

No infinite antichains.

- Words over a finite alphabet [Higman].
- Graphs closed under minors [Robertson and Seymour].

Infinite antichains.

- Graphs closed under induced subgraphs (or merely subgraphs). e.g. $C_3, C_4, C_5, \ldots$
- Permutations closed under containment.
- Tournaments, digraphs, ...

Partial Well Order

- A permutation class is partially well-ordered (pwo) if it contains no infinite antichains.

Partial Well Order

- A permutation class is partially well-ordered (pwo) if it contains no infinite antichains.

Question

Can we decide whether a permutation class given by a finite basis is pwo?

- To prove pwo — Higman's theorem is useful.
- To prove not pwo — find an antichain.

Partial Well Order

- A permutation class is partially well-ordered (pwo) if it contains no infinite antichains.

Question

Can we decide whether a hereditary property given by a finite basis is wqo?

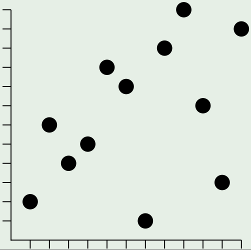
- To prove pwo — Higman's theorem is useful.
- To prove not pwo — find an antichain.
- Other structures: well quasi-order, not pwo, but same idea.

Outline

Intervals

- Pick any permutation π.
- An interval of π is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.

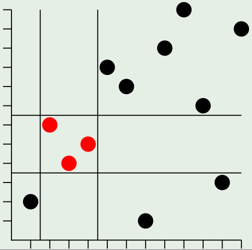
Example



Intervals

- Pick any permutation π.
- An interval of π is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.

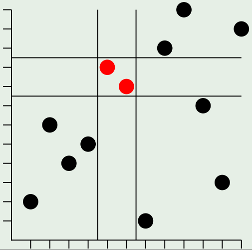
Example



Intervals

- Pick any permutation π.
- An interval of π is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.

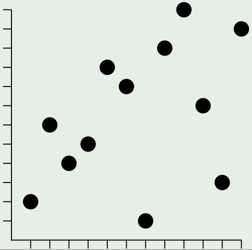
Example



Intervals

- Pick any permutation π.
- An interval of π is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.
- Intervals are important in biomathematics (genetic algorithms, matching gene sequences).

Example



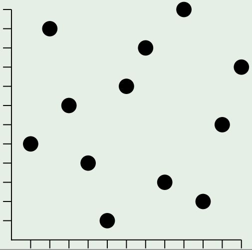
Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

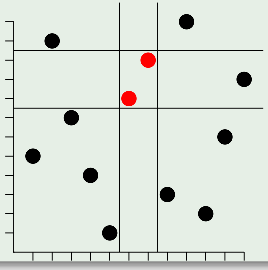
Example



Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

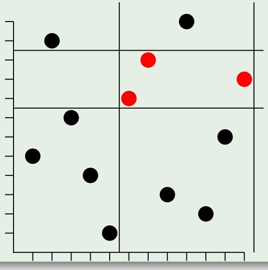
Example



Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

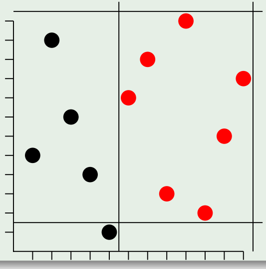
Example



Simple Permutations

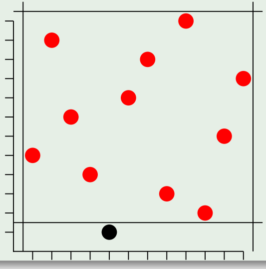
- A simple permutation: The only intervals are singletons and the whole thing.

Example



- A simple permutation: The only intervals are singletons and the whole thing.

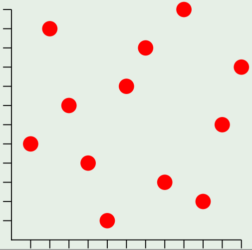
Example



Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

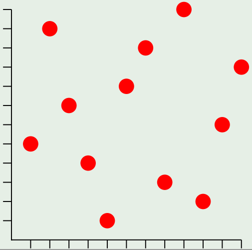
Example



Simple Permutations

- A simple permutation: The only intervals are singletons and the whole thing.

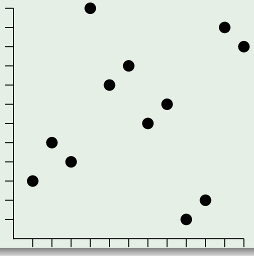
Example



- 1 is simple, as are 12 and 21.
- There are no simple permutations of length three.
- Two of length four: 2413 and 3142.

- Simple permutations are the “building blocks” of all permutations.

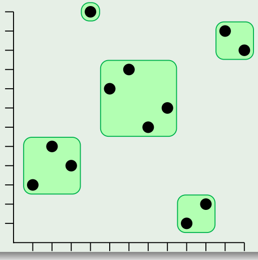
Example



Decomposing Permutations

- Simple permutations are the "building blocks" of all permutations.
- Break permutation into maximal proper intervals.

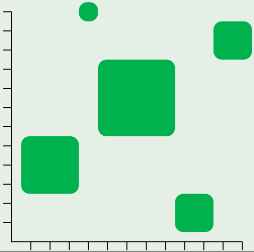
Example



Decomposing Permutations

- Simple permutations are the "building blocks" of all permutations.
- Gives a unique simple permutation, the skeleton.

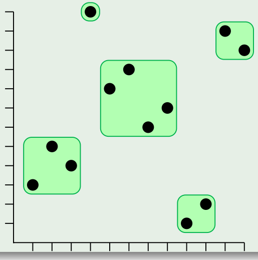
Example



Decomposing Permutations

- Simple permutations are the "building blocks" of all permutations.
- If simple has > 2 points then the blocks are unique.

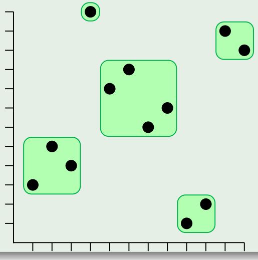
Example



Decomposing Permutations

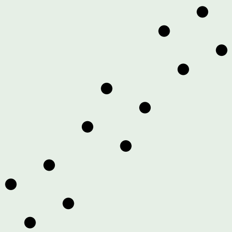
- Simple permutations are the “building blocks” of all permutations.
- If simple has > 2 points then the blocks are unique.
- This decomposition is the substitution decomposition.

Example



- Simple permutation of length 2: block decomposition is not unique.

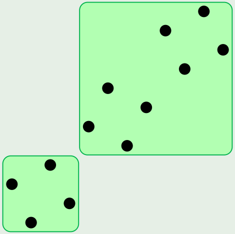
Example



Non-uniqueness

- Simple permutation of length 2: block decomposition is not unique.

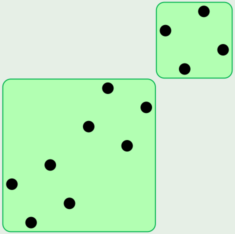
Example



Non-uniqueness

- Simple permutation of length 2: block decomposition is not unique.

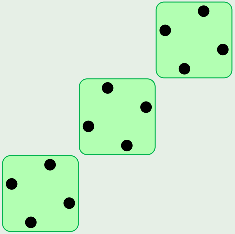
Example



Non-uniqueness

- Underlying structure is an increasing permutation.

Example



Finitely Many Simples

Using the substitution decomposition, we can say a lot about permutation classes that contain only finitely many simples [Albert and Atkinson, 2005]:

Finitely Many Simples

Using the substitution decomposition, we can say a lot about permutation classes that contain only finitely many simples [Albert and Atkinson, 2005]:

- They have a finite basis.
- They are enumerated by algebraic generating functions.
- They are partially well-ordered.

Finitely Many Simples

Using the substitution decomposition, we can say a lot about permutation classes that contain only finitely many simples [Albert and Atkinson, 2005]:

- They have a finite basis.
- They are enumerated by algebraic generating functions.
- They are partially well-ordered.

Theorem (B., Ruškuc and Vatter, 2008)

It is possible to decide whether a permutation class given by a finite basis contains infinitely many simple permutations.

Finitely Many Simples

Using the substitution decomposition, we can say a lot about permutation classes that contain only finitely many simples [Albert and Atkinson, 2005]:

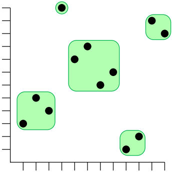
- They have a finite basis.
- They are enumerated by algebraic generating functions.
- They are partially well-ordered.

Theorem (B., Ruškuc and Vatter, 2008)

It is possible to decide whether a permutation class given by a finite basis contains infinitely many simple permutations.

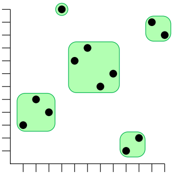
- There should be a graph-theoretic analogue of this result!

Finitely Many Simples ⇒ Partially Well-Ordered



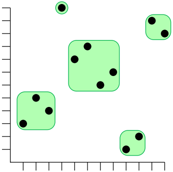
- Take a class $\mathcal{C}$ containing a finite set S of simple permutations.
- Every permutation in $\mathcal{C}$ has a skeleton from S.

Finitely Many Simples ⇒ Partially Well-Ordered



- Take a class $\mathcal{C}$ containing a finite set S of simple permutations.
- Every permutation in $\mathcal{C}$ has a skeleton from S.
- Think of each $\sigma \in S$ of length n as an n-ary operation.
- Starting with the permutation 1, we build every permutation in the class $\mathcal{C}$ by recursively using this finite set of operations.

Finitely Many Simples ⇒ Partially Well-Ordered



- Take a class $\mathcal{C}$ containing a finite set S of simple permutations.
- Every permutation in $\mathcal{C}$ has a skeleton from S.
- Think of each $\sigma \in S$ of length n as an n-ary operation.
- Starting with the permutation 1, we build every permutation in the class $\mathcal{C}$ by recursively using this finite set of operations.
- Now use Higman's Theorem.

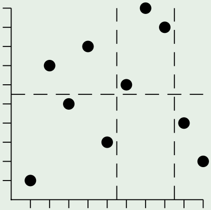
Outline

Grid Classes

- Matrix $\mathcal{M}$ whose entries are permutation classes.
- $\mathrm{Grid}(\mathcal{M})$ the grid class of $\mathcal{M}$: all permutations which can be "gridded" so each cell satisfies constraints of $\mathcal{M}$.

Example

- Let $\mathcal{M} = \begin{pmatrix} \mathrm{Av}(21) & \mathrm{Av}(231) & \emptyset \\ \mathrm{Av}(123) & \emptyset & \mathrm{Av}(12) \end{pmatrix}$.



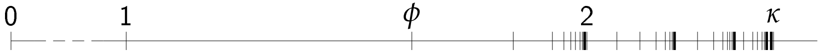
$\in \mathrm{Grid}(\mathcal{M})$

Grid classes are useful

- Recall: Growth rate of $\mathcal{C}$ is $\lim_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$ (if it exists).

Grid classes are useful

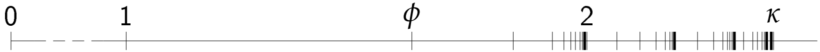
- Recall: Growth rate of $\mathcal{C}$ is $\lim_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$ (if it exists).
- Using grid classes: Below $\kappa \approx 2.20557$, growth rates exist and can be characterised [Vatter]:



- κ is the lowest growth rate where we encounter infinite antichains, and hence uncountably many permutation classes.

Grid classes are useful

- Recall: Growth rate of $\mathcal{C}$ is $\lim_{n\to\infty} \sqrt[n]{|\mathcal{C}_n|}$ (if it exists).
- Using grid classes: Below $\kappa \approx 2.20557$, growth rates exist and can be characterised [Vatter]:



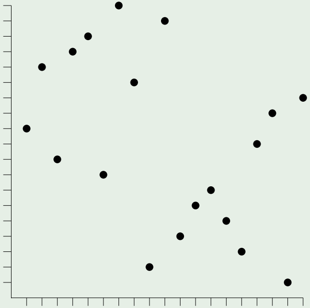
- κ is the lowest growth rate where we encounter infinite antichains, and hence uncountably many permutation classes.
- Cf "canonical properties" of graphs [Balogh, Bollobás and Weinreich].

Monotone Grid Classes

- Special case: all cells of $\mathcal{M}$ are $\mathrm{Av}(21)$ or $\mathrm{Av}(12)$.
- Rewrite $\mathcal{M}$ as a matrix with entries in $\{0, 1, -1\}$.

Example

$$\mathcal{M} = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}$$

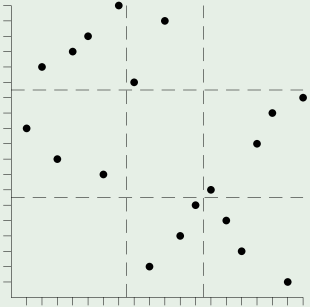


Monotone Grid Classes

- Special case: all cells of $\mathcal{M}$ are Av(21) or Av(12).
- Rewrite $\mathcal{M}$ as a matrix with entries in $\{0, 1, -1\}$.

Example

$$\mathcal{M} = \begin{pmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}$$



The Graph of a Matrix

- Graph of a matrix, $G(\mathcal{M})$, formed by connecting together all non-zero entries that share a row or column and are not "separated" by any other nonzero entry.

Example

$$\begin{pmatrix} \mathcal{C} & 0 & 0 & \mathcal{D} \\ 0 & 0 & \mathcal{E} & 0 \\ \mathcal{D} & \mathcal{E} & 0 & \mathcal{C} \\ 0 & 0 & 0 & \mathcal{D} \end{pmatrix}$$

The Graph of a Matrix

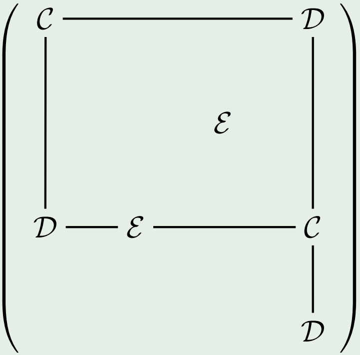
- Graph of a matrix, $G(\mathcal{M})$, formed by connecting together all non-zero entries that share a row or column and are not “separated” by any other nonzero entry.

Example

$$\begin{pmatrix} \mathcal{C} & & & \mathcal{D} \\ & & \mathcal{E} & \\ \mathcal{D} & \mathcal{E} & & \mathcal{C} \\ & & & \mathcal{D} \end{pmatrix}$$

- Graph of a matrix, $G(\mathcal{M})$, formed by connecting together all non-zero entries that share a row or column and are not “separated” by any other nonzero entry.

Example



Monotone Grids and Partial Well-Order

Theorem (Murphy and Vatter, 2003)

The monotone grid class $\mathrm{Grid}(\mathcal{M})$ *is pwo if and only if* $G(\mathcal{M})$ *is a forest, i.e.* $G(\mathcal{M})$ *contains no cycles.*

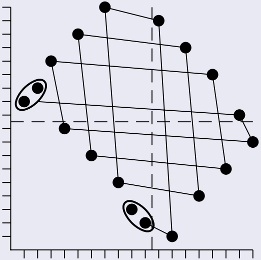
Monotone Grids and Partial Well-Order

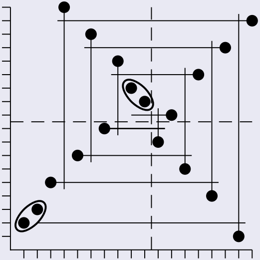
Theorem (Murphy and Vatter, 2003)

The monotone grid class $\mathrm{Grid}(\mathcal{M})$ *is pwo if and only if* $G(\mathcal{M})$ *is a forest, i.e.* $G(\mathcal{M})$ *contains no cycles.*

Proof.

($\Rightarrow$) Construct infinite antichains that "walk" around a cycle.





□

When does that apply?

Question

When is a class $\mathcal{C}$ (a subset of) a monotone grid class?

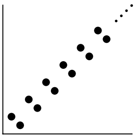
When does that apply?

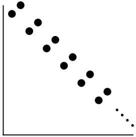
Question

When is a class $\mathcal{C}$ (a subset of) a monotone grid class?

Answer [Vatter]

A class $\mathcal{C}$ is monotone griddable if and only if it contains neither the classes $\oplus 21$ nor $\ominus 12$.





Non-monotone cells

- If a class is not monotone griddable, then perhaps it can be gridded by a matrix which is mostly monotone:

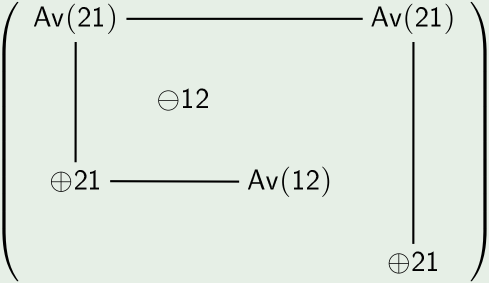
Example

$$\begin{pmatrix} \mathrm{Av}(21) & 0 & 0 & \mathrm{Av}(21) \\ 0 & \ominus 12 & 0 & 0 \\ \oplus 21 & 0 & \mathrm{Av}(12) & 0 \\ 0 & 0 & 0 & \oplus 21 \end{pmatrix}$$

Non-monotone cells

- If a class is not monotone griddable, then perhaps it can be gridded by a matrix which is mostly monotone:

Example



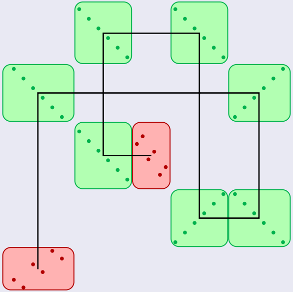
- To be pwo, graph must still be a forest, but now the number of non-monotone-griddable cells in each component matters.

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.

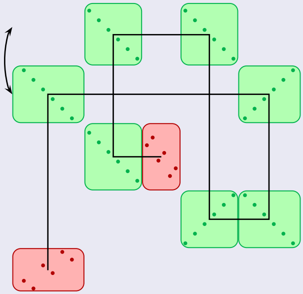
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.

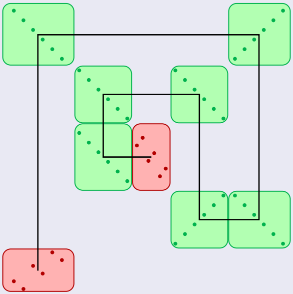
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.

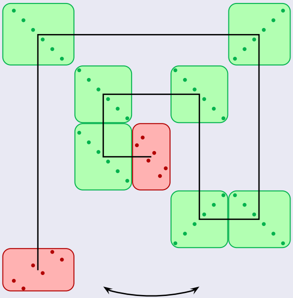
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.

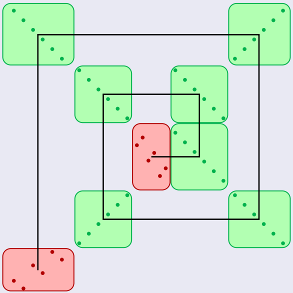
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.

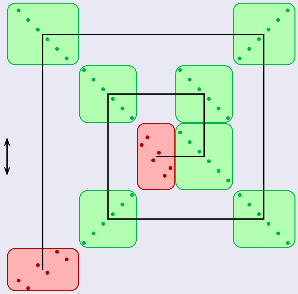
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.

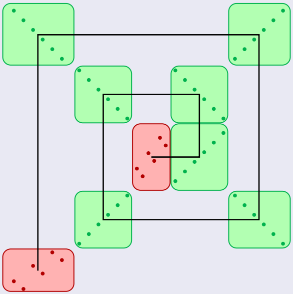
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.

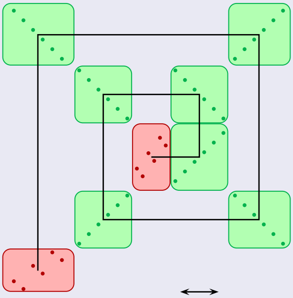
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.

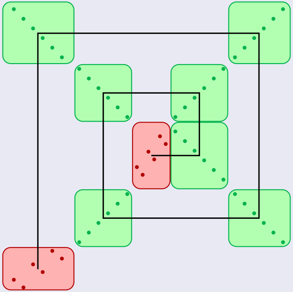
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.

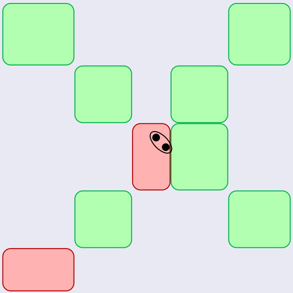
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

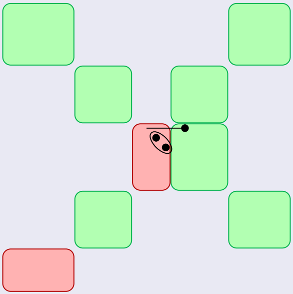
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

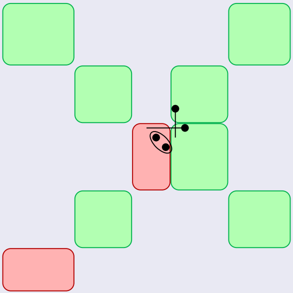
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

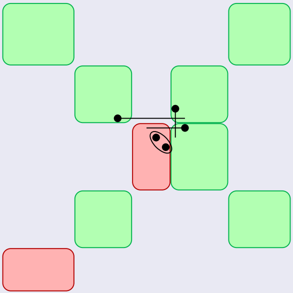
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

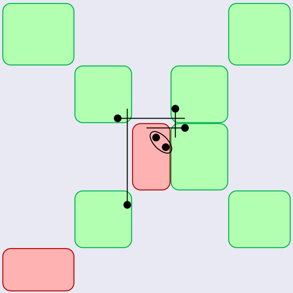
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

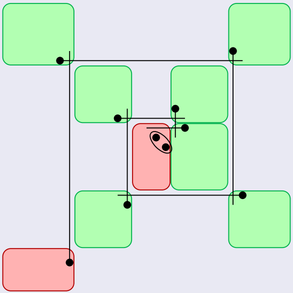
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

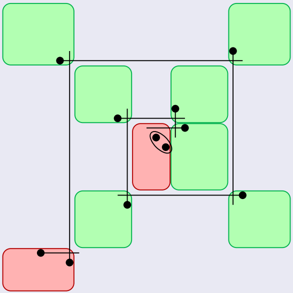
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

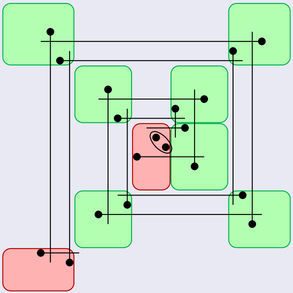
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

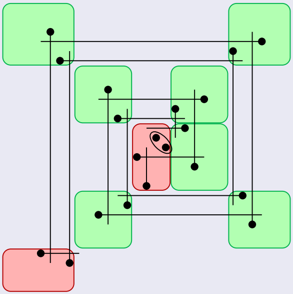
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

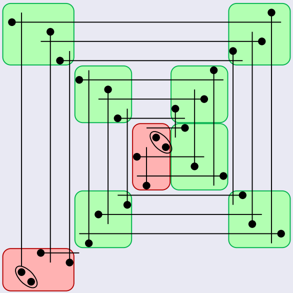
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

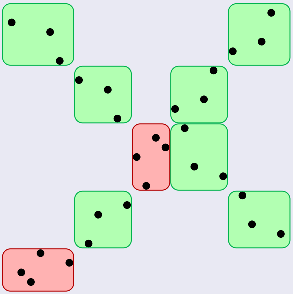
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.

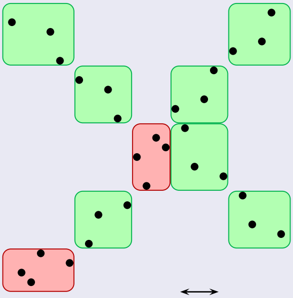
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

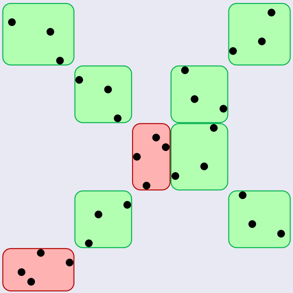
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

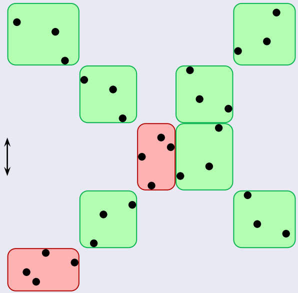
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

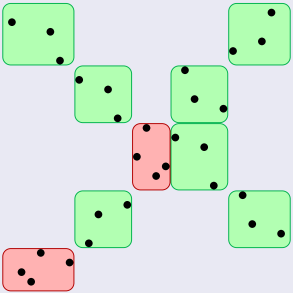
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

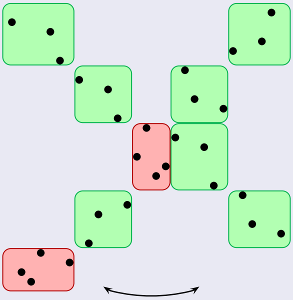
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

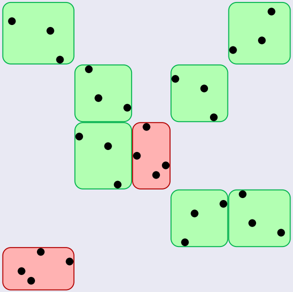
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

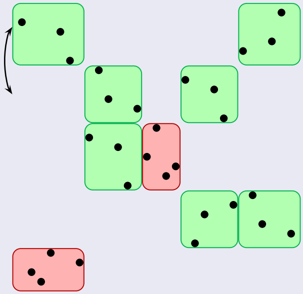
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.



- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.

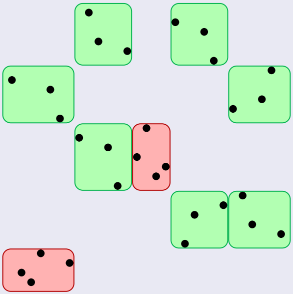
□

Two is too many

Theorem

A grid class whose graph has a component containing two or more non-monotone-griddable classes is not pwo.

Proof.

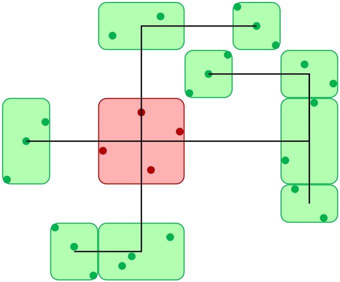


- WLOG graph is a path connecting two bad cells.
- Permute rows and columns.
- Flip rows and columns.
- Build antichain with grid pin sequences.
- Flip and permute back.
- Still have an antichain.

□

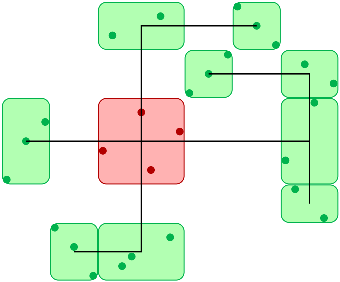
Just one non-monotone

- Suppose the bad cell contains only finitely many simple permutations.



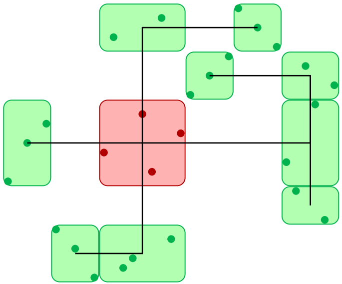
Just one non-monotone

- Suppose the bad cell contains only finitely many simple permutations.
- Build permutations component-wise: use the substitution decomposition on the red cell, and view each component as a tree rooted on this cell.



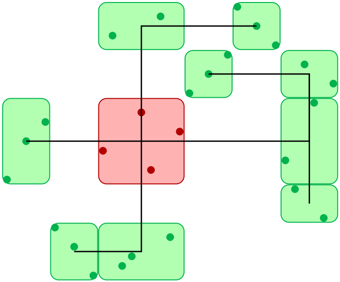
Just one non-monotone

- Suppose the bad cell contains only finitely many simple permutations.
- Build permutations component-wise: use the substitution decomposition on the red cell, and view each component as a tree rooted on this cell.
- This defines a construction for all permutations in the grid class, which is amenable to Higman's Theorem.



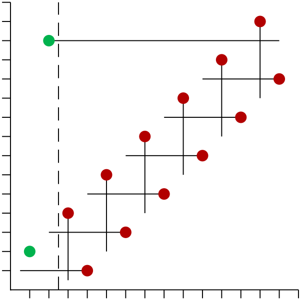
Theorem

Let $\mathcal{M}$ be a gridding matrix for which each component is a forest and contains at most one non-monotone cell. If every non-monotone cell contains only finitely many simple permutations, then $\mathrm{Grid}(\mathcal{M})$ is pwo.



But sometimes one is too much...

- One cell containing arbitrarily long increasing oscillations next to a monotone cell is bad...



Outline

Summary

- Two non-monotone per component: class not pwo.
- One non-monotone but finitely many simples: class is pwo.

Summary

- Two non-monotone per component: class not pwo.
- One non-monotone but finitely many simples: class is pwo.
- To-do: one non-monotone but infinitely many simples (some antichains known).

Summary

- Two non-monotone per component: class not pwo.
- One non-monotone but finitely many simples: class is pwo.
- To-do: one non-monotone but infinitely many simples (some antichains known).

Question

Can we decide whether a permutation class given by a finite basis is pwo?

- There are still a lot of obstacles, but maybe we're a bit closer.

Thanks!