Antichains of Permutations

Robert Brignall

Department of Mathematics
University of Bristol

Tuesday 17th June, 2008
1. Introduction
   - Permutation Classes
   - Antichains
   - Partial Well Order

2. Grid Classes
   - Monotone Classes
   - Antichains and Pin Sequences
   - Juxtapositions
A permutation \( \tau = t_1 t_2 \ldots t_k \) is contained in the permutation \( \sigma = s_1 s_2 \ldots s_n \) if there exists a subsequence \( s_{i_1}, s_{i_2}, \ldots, s_{i_k} \) order isomorphic to \( \tau \).
A permutation $\tau = t_1 t_2 \ldots t_k$ is contained in the permutation $\sigma = s_1 s_2 \ldots s_n$ if there exists a subsequence $s_{i_1}, s_{i_2}, \ldots, s_{i_k}$ order isomorphic to $\tau$. 

**Example**

\[\begin{align*}
1 & 3 & 5 & 2 & 4 \\
4 & 2 & 1 & 6 & 3 & 8 & 5 & 7
\end{align*}\]
A permutation $\tau = t_1 t_2 \ldots t_k$ is contained in the permutation $\sigma = s_1 s_2 \ldots s_n$ if there exists a subsequence $s_{i_1}, s_{i_2}, \ldots, s_{i_k}$ order isomorphic to $\tau$.

Example

- $1 3 5 2 4 \prec 4 2 1 6 3 8 5 7$
Containment forms a **partial order** on the set of all permutations.
Containment forms a partial order on the set of all permutations.

Downsets of permutations in this partial order form permutation classes.

i.e. \( \pi \in C \) and \( \sigma \leq \pi \) implies \( \sigma \in C \).
Containment forms a **partial order** on the set of all permutations.

Downsets of permutations in this partial order form permutation classes.

i.e. $\pi \in C$ and $\sigma \leq \pi$ implies $\sigma \in C$.

A permutation class $C$ can be seen to avoid certain permutations. Write $C = \text{Av}(B) = \{\pi : \beta \not\leq \pi \text{ for all } \beta \in B\}$.
Containment forms a **partial order** on the set of all permutations.

Downsets of permutations in this partial order form permutation classes.

i.e. \(\pi \in C\) and \(\sigma \leq \pi\) implies \(\sigma \in C\).

A permutation class \(C\) can be seen to avoid certain permutations. Write \(C = \text{Av}(B) = \{\pi : \beta \not\leq \pi \text{ for all } \beta \in B\}\).

The minimal avoidance set is the **basis**. It is **unique** but need not be finite.
Containment forms a partial order on the set of all permutations.

Downsets of permutations in this partial order form permutation classes.

i.e. \( \pi \in C \) and \( \sigma \leq \pi \) implies \( \sigma \in C \).

A permutation class \( C \) can be seen to avoid certain permutations. Write \( C = Av(B) = \{ \pi : \beta \not\preceq \pi \text{ for all } \beta \in B \} \).

The minimal avoidance set is the basis. It is unique but need not be finite.

**Example**

The class \( C = Av(12) \) consists of all the decreasing permutations:

\( \{1, 21, 321, 4321, \ldots\} \)
Antichains

- Set of *pairwise incomparable* permutations.
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

- **Bottom** copies of 4123 must match up (the anchor).
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

- Each point is matched in turn.
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

- Each point is matched in turn.
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

- Each point is matched in turn.
Antichains

- Set of pairwise incomparable permutations.

Example (Increasing Oscillating Antichain)

- Last pair cannot be embedded.
Complete and Fundamental Antichains

- **Closure** of a set $A$: $\text{Cl}(A) = \{\pi : \pi \leq \alpha \text{ for some } \alpha \in A\}$. 
Closure of a set $A$: $\text{Cl}(A) = \{ \pi : \pi \leq \alpha \text{ for some } \alpha \in A \}$.

An infinite antichain $A$ is fundamental if $\text{Cl}(A)$ contains no infinite antichains except for $A$ and its subsets.
Complete and Fundamental Antichains

- **Closure of a set** $A$: $\text{Cl}(A) = \{ \pi : \pi \leq \alpha \text{ for some } \alpha \in A \}$.

- An infinite antichain $A$ is **fundamental** if $\text{Cl}(A)$ contains no infinite antichains except for $A$ and its subsets.

- An infinite antichain is **complete** if no permutation can be added to make a bigger antichain.
Complete and Fundamental Antichains

- An infinite antichain \( A \) is **fundamental** if \( \text{Cl}(A) \) contains no infinite antichains except for \( A \) and its subsets.
- An infinite antichain is **complete** if no permutation can be added to make a bigger antichain.

**Example**

The increasing oscillating antichain is fundamental, but not complete.

**Not complete:** \( I \cup \{321\} \) is an antichain.
For any permutation $\pi$ and antichain $A$, $A^{||\pi} = \{\alpha \in A : \pi^{||}\alpha\}$. 
For any permutation $\pi$ and antichain $A$, $A^{\parallel \pi} = \{ \alpha \in A : \pi \parallel \alpha \}$.

**Lemma**

A is fundamental if and only if the proper closure $Cl(A) \setminus A$ is pwo and for every $\pi \in Cl(A) \setminus A$ the set $A^{\parallel \pi}$ is finite.
For any permutation $\pi$ and antichain $A$, $A^{||\pi} = \{\alpha \in A : \pi^{||\alpha}\}$.

**Lemma**

A is fundamental if and only if the proper closure $\text{Cl}(A) \setminus A$ is pwo and for every $\pi \in \text{Cl}(A) \setminus A$ the set $A^{||\pi}$ is finite.

This condition means that terms of a fundamental antichain look “similar”.
Conjecture (Murphy)

*If A is a fundamental antichain then there exist only finitely many lengths n such that A has two or more permutations of length n.*
Conjecture (Murphy)

If A is a fundamental antichain then there exist only finitely many lengths n such that A has two or more permutations of length n.

Conjecture

Every member of a fundamental antichain contains at most two proper intervals.
Define an order on antichains:

\[ B \leq A \iff \text{for every } \alpha \in A, \text{ there exists } \beta \in B \text{ with } \beta \leq \alpha \]

Note that \( A \subseteq B \) implies \( B \leq A \)!

Interested in antichains that are \textit{minimal} under \( \leq \).
An Ordering on Antichains

- Define an order on antichains:

  \[ B \leq A \iff \text{for every } \alpha \in A, \text{ there exists } \beta \in B \text{ with } \beta \leq \alpha \]

- Note that \( A \subseteq B \) implies \( B \leq A \)!

- Interested in antichains that are **minimal** under \( \leq \).

**Lemma**

An antichain is minimal under \( \leq \) if and only if it is complete and fundamental.
A permutation class is partially well-ordered (pwo) if it contains no infinite antichains.
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.
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.

**Proposition (Nash-Williams, 1963)**

Every non-pwo permutation class contains an antichain that is minimal under $\preceq$.

**Corollary**

Every non-pwo permutation class contains a fundamental antichain.
Theorem (Cherlin and Latka, 2000)

For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.
Theorem (Cherlin and Latka, 2000)

For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.

- For hereditary properties of tournaments, $\Lambda_1$ has been identified.
More on Minimal Antichains

Theorem (Cherlin and Latka, 2000)

For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.

For hereditary properties of tournaments, $\Lambda_1$ has been identified.

Proposition (Cherlin and Latka)

The problem of deciding whether a hereditary property of tournaments with two basis elements is pwo is decidable in polynomial time.
More on Minimal Antichains

**Theorem (Cherlin and Latka, 2000)**

For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.

- For hereditary properties of tournaments, $\Lambda_1$ has been identified.

**Proposition (Cherlin and Latka)**

The problem of deciding whether a hereditary property of tournaments with two basis elements is pwo is decidable in polynomial time.

- Caveat: algorithm is not known.
Theorem (Cherlin and Latka, 2000)

For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.

- For permutation classes, $\Lambda_1$ consists of the minimal antichains containing increasing oscillating, Widdershins and $V$. 
More on Minimal Antichains

**Theorem (Cherlin and Latka, 2000)**

*For each natural number $k$, there is a finite set $\Lambda_k$ of antichains minimal under $\preceq$ with the property that a class avoiding exactly $k$ permutations is pwo if and only if its intersection with each antichain in $\Lambda_k$ is finite.*

- For permutation classes, $\Lambda_1$ consists of the minimal antichains containing increasing oscillating, Widdershins and $V$.

**Proposition (Atkinson, Murphy and Ruškuc, 2002)**

$\text{Av}(\beta)$ is pwo if and only if $\beta \in \{1, 12, 21, 132, 213, 231, 312\}$
• \( \mathcal{A} \) — set of all minimal antichains, viewed as a topological space.
• **Open sets**: for \( B \) a finite set of permutations

\[ \mathcal{A}_B = \{ A \in \mathcal{A} : A \cap \text{Av}(B) \text{ is infinite} \} . \]
Topology

- $\mathcal{A}$ — set of all minimal antichains, viewed as a topological space.
- **Open sets**: for $B$ a finite set of permutations
  \[ \mathcal{A}_B = \{ A \in \mathcal{A} : A \cap \text{Av}(B) \text{ is infinite} \}. \]

- **Equivalence relation**:
  \[ A_1 \rho A_2 \iff \{ \mathcal{A}_B : A_1 \in \mathcal{A}_B \} = \{ \mathcal{A}_B : A_1 \in \mathcal{A}_B \}. \]

- Easier: $A_1 \rho A_2$ iff $\text{Cl}(A_1) \setminus A = \text{Cl}(A_2) \setminus A$.
- **Quotient**: $\mathcal{A}' = \mathcal{A}/\rho$ (is a $T_0$ space).
Topology

- $\mathcal{A}$ — set of all minimal antichains, viewed as a topological space.
- **Open sets:** for $B$ a finite set of permutations
  \[
  \mathcal{A}_B = \{ A \in \mathcal{A} : A \cap \text{Av}(B) \text{ is infinite}\}.
  \]
- **Equivalence relation:**
  \[
  A_1 \rho A_2 \Leftrightarrow \{ \mathcal{A}_B : A_1 \in \mathcal{A}_B \} = \{ \mathcal{A}_B : A_1 \in \mathcal{A}_B \}.
  \]
- Easier: $A_1 \rho A_2$ iff $\text{Cl}(A_1) \setminus A = \text{Cl}(A_2) \setminus A$.
- **Quotient:** $\mathcal{A}' = \mathcal{A}/\rho$ (is a $T_0$ space).
- $A \in \mathcal{A}$ is isolated in $\mathcal{A}'$ if there is some finite basis $B$ such all infinite fundamental antichains in $\text{Av}(B)$ are equivalent (in $\mathcal{A}'$) to $A$. 
Cherlin and Latka asked these for tournaments, but why not ask them for permutations?
Conjectures

- Cherlin and Latka asked these for tournaments, but why not ask them for permutations?

Conjecture

Not all minimal antichains are isolated.

- There are some minimal antichains that are never needed to prove that a finitely based class is non-pwo.
Conjectures

• Cherlin and Latka asked these for tournaments, but why not ask them for permutations?

Conjecture

Not all minimal antichains are isolated.

• There are some minimal antichains that are never needed to prove that a finitely based class is non-pwo.

Conjecture

For each isolated antichain $A$ “in” $A'$, there is an algorithm to decide whether an arbitrary permutation belongs to $C_l(A) \setminus A$.

• Minimal isolated antichains have some kind of reliable structure.
Grid Classes

- **Matrix** $\mathcal{M}$ whose entries are permutation classes.
- **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 $\mathcal{M} = \begin{pmatrix} \text{Av}(21) & \text{Av}(231) & \emptyset \\ \text{Av}(123) & \emptyset & \text{Av}(12) \end{pmatrix}$. 

\[\in \text{Grid} (\mathcal{M})\]
Monotone Grid Classes

- **Special case**: all cells of $\mathcal{M}$ are $\text{Av}(21)$ or $\text{Av}(12)$.
- 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 $\text{Av}(21)$ or $\text{Av}(12)$.
- 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**, $G(M)$, formed by connecting together all non-zero entries that share a row or column and are not “separated” by any other nonzero entry.

### Example

$$
\begin{pmatrix}
1 & 0 & 0 & 0 & -1 \\
0 & 0 & 1 & 0 & 0 \\
-1 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & -1
\end{pmatrix}
$$
The Graph of a Matrix

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

**Example**

\[
\begin{pmatrix}
1 & -1 \\
-1 & -1 & 1 \\
-1 & 1 \\
\end{pmatrix}
\]
The Graph of a Matrix

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

Example

\[
\begin{pmatrix}
1 & -1 & 1 \\
-1 & -1 & 1 \\
1 & 1 & -1
\end{pmatrix}
\]
Theorem (Murphy and Vatter, 2003)

The monotone grid class $\text{Grid}(\mathcal{M})$ is pwo if and only if $G(\mathcal{M})$ is a forest, i.e. $G(\mathcal{M})$ contains no cycles.
Theorem (Murphy and Vatter, 2003)

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

Proof.

Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ Partial multiplication table.
Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ $\pm 1$ correspond to directions.

\[ \begin{array}{c|c|c|c}
\hline
1 & -1 & & \\
\hline
& 1 & & \\
\hline
-1 & & -1 & \\
\hline
\end{array} \]
The monotone grid class $\text{Grid}(\mathcal{M})$ is pwo if and only if $G(\mathcal{M})$ is a forest, i.e. $G(\mathcal{M})$ contains no cycles.

Proof.

$(\Leftarrow)$ Form one order per arrow.

- $1 < 9 < 8 < 4$.
- $5 < 10 < 6 < 7$.
- $2 < 3$.
- $1 < 2 < 3 < 4$.
- $5 < 6$.
- $10 < 9 < 8 < 7$. 
Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ No cycles, so this gives a poset.

- $1 < 9 < 8 < 4$.
- $5 < 10 < 6 < 7$.
- $2 < 3$.
- $1 < 2 < 3 < 4$.
- $5 < 6$.
- $10 < 9 < 8 < 7$. 

(University of Bristol)
Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ Linear extension: $5 < 10 < 1 < 9 < 2 < 6 < 8 < 3 < 7 < 4$

\[
\begin{array}{c}
5 \\
10 \\
1 \\
9 \\
8 \\
4 \\
2 \\
3 \\
7 \\
6
\end{array}
\]
Monotone Grids and Partial Well Order

Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ Linear extension: $5 < 10 < 1 < 9 < 2 < 6 < 8 < 3 < 7 < 4$

Encode by region: 3412532541.
Monotone Grids and Partial Well Order

Theorem (Murphy and Vatter, 2003)

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

Proof.

$(\Leftarrow)$ Linear extension: $5 < 10 < 1 < 9 < 2 < 6 < 8 < 3 < 7 < 4$

- Encode by region: $3412532541$.
- Higman's Theorem: $\{1, 2, 3, 4, 5\}^*$ is pwo under the subword order.
- Encoding is reversible, hence $\text{Grid}(\mathcal{M})$ is pwo.
Theorem (Murphy and Vatter, 2003)

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

Proof.

\( (\Rightarrow) \) Construct fundamental antichains that “walk” around a cycle.
The Widdershins Antichain

“Spirals” out from the centre.

Constructed by means of a **pin sequence**.

In general: a pin sequence with first and last pins inflated forms a fundamental antichain.
Quasi-Square

- Not constructible by a pin sequence.
Quasi-Square

- Not constructible by a pin sequence.
- Flip each column...
Quasi-Square

- Not constructible by a pin sequence.
- ...Widdershins!
Bigger Grids

Carry out row flips and column reversals: \( r_1 \circ r_2 \circ r_3 \circ f_3 \).
Carry out row flips and column reversals: \( r_1 \circ r_2 \circ r_3 \circ f_3 \).
Carry out row flips and column reversals: \( r_1 \circ r_2 \circ r_3 \circ f_3 \).
Carry out row flips and column reversals: $r_1 \circ r_2 \circ r_3 \circ f_3$. 
Carry out row flips and column reversals: $r_1 \circ r_2 \circ r_3 \circ f_3$. 
Carry out row flips and column reversals: $r_1 \circ r_2 \circ r_3 \circ f_3$. 
Bigger Grids

Carry out row flips and column reversals: $r_1 \circ r_2 \circ r_3 \circ f_3$. 

Resulting structure behaves a bit like a pin sequence.
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

**Example**

![Diagram showing grid pin sequences with pins labeled $p_1$ and $p_2$.](image-url)
Grid Pin Sequences

- **Local separation**: \( p_{i+1} \) separates \( p_i \) from \( p_{i+1} \).
- **Row-column agreement**: \( p_{i+1} \) must be placed in the same row or column as \( p_i \).
- **Local externality**: \( p_{i+1} \) extends from \( \text{Rect}(p_{i-1}, p_i) \).
- **Non-interaction**: \( p_{i+1} \) could not have been used in \( p_1, \ldots, p_i \).

**Example**

![Diagram showing grid pin sequences with points labeled \( p_1, p_2, p_3 \).]
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

### Example

![Diagram of grid pin sequences with points $p_2$, $p_3$, $p_4$]
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

---

**Example**

![Diagram](image_url)

Example of local separation, row-column agreement, and local externality.
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

Example
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

**Example**
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_{i-1}, p_i)$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

Example

```
\begin{align*}
\text{(University of Bristol)}
\end{align*}
```
Grid Pin Sequences

- **Local separation**: $p_{i+1}$ separates $p_i$ from $p_{i+1}$.
- **Row-column agreement**: $p_{i+1}$ must be placed in the same row or column as $p_i$.
- **Local externality**: $p_{i+1}$ extends from $\text{Rect}(p_i, p_{i-1})$.
- **Non-interaction**: $p_{i+1}$ could not have been used in $p_1, \ldots, p_i$.

Example
Grid Pin Sequences

- **Local separation**: \( p_{i+1} \) separates \( p_i \) from \( p_{i+1} \).
- **Row-column agreement**: \( p_{i+1} \) must be placed in the same row or column as \( p_i \).
- **Local externality**: \( p_{i+1} \) extends from \( \text{Rect}(p_{i-1}, p_i) \).
- **Non-interaction**: \( p_{i+1} \) could not have been used in \( p_1, \ldots, p_i \).

**Example**

![Example diagram showing grid pin sequences](image)
Grid pin sequences on an $m \times n$ grid can be encoded in a regular language on \( \{ c_1, \ldots, c_m, r_1, \ldots, r_n \} \).
Grid pin sequences on an $m \times n$ grid can be **encoded in a regular language** on $\{c_1, \ldots, c_m, r_1, \ldots, r_n\}$.

**Monotone grid classes** — we only need to check grid pin sequences that go round in “circles”.
Optimism

- Grid pin sequences on an $m \times n$ grid can be encoded in a regular language on \{${c_1, \ldots, c_m, r_1, \ldots, r_n}$\}.
- Monotone grid classes — we only need to check grid pin sequences that go round in “circles”.

Conjecture

*It is decidable whether a subclass of monotone grid class (“monotone griddable”) given by a finite basis is partially well ordered.*
Grid pin sequences on an $m \times n$ grid can be encoded in a regular language on $\{c_1, \ldots, c_m, r_1, \ldots, r_n\}$.

Monotone grid classes — we only need to check grid pin sequences that go round in “circles”.

**Conjecture**

*It is decidable whether a subclass of monotone grid class ("monotone griddable") given by a finite basis is partially well ordered.*

**Theorem (Hucznyska and Vatter, 2006)**

*A permutation class is monotone griddable if and only if it does not contain arbitrarily long sums of 21 or skew sums of 12.*
Other Antichains

- Increasing Oscillating — pin sequence in a single cell.

![Graph showing increasing oscillating pin sequence in a single cell.](image-url)
Two cells: antichain $V$. 
Other Antichains

- Two cells: antichain $V$.
- LHS: skew sums of 12.
Two cells: antichain $V$.
RHS: direct sums of 21.
The **juxtaposition** of two classes $\mathcal{C}$ and $\mathcal{D}$ is $[\mathcal{C} \: \mathcal{D}] = \text{Grid}(\mathcal{C} \: \mathcal{D})$.

Think of them as grid classes with 2 cells.

**Question**

*When is the juxtaposition of two classes pwo?*
If $D$ contains arbitrarily long oscillations and $C \neq \text{Av}(12, 21)$ then $[C \ D]$ is not pwo. (“Tied by One” antichain)
If $C$ and $D$ both contain arbitrarily long sums of 21 or skew sums of 12, then $[C \ D]$ is not pwo.
If $\mathcal{C}$ and $\mathcal{D}$ do not contain arbitrarily long sums of 21 or skew sums of 12, then they are monotone griddable.

Not pwo if $\mathcal{C}$ and $\mathcal{D}$ contain arbitrarily long vertical alternations.
Thanks!
Appendix: Proper Pin Sequences

Start with a point placed in relation to the origin.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making \(90^\circ\) turns.
Appendix: Proper Pin Sequences

- Start with a point placed in relation to the origin.
- Extend up, down, left, or right – this is an up pin.
- A proper pin must be external and cut the previous pin, but not the rectangle.
- Proper pins must travel by making $90^\circ$ turns.
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2$

Example

\[ \begin{array}{cccc}
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
\end{array} \]

\[ r_3 \quad r_2 \quad r_1 \]

\[ \begin{array}{ccc}
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
   \text{\_\_\_} & \text{\_\_\_} & \text{\_\_\_} \\
\end{array} \]

\[ c_1 \quad c_2 \quad c_3 \]
Letter $r_i$: place a pin in row $i$.
Letter $c_j$: place a pin in column $j$.
This defines the placement of the pin uniquely.
For example: $r_2c_3$
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2c_3r_2$

Example

```
  r_3 __________
  |          |
  r_2       |
  |          |
  r_1|________|
     |     |
     c_1 c_2 c_3
```
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2c_3r_2c_1$

**Example**

![Example Diagram]

(University of Bristol) Antichains of Permutations
Letter $r_i$: place a pin in row $i$.
Letter $c_j$: place a pin in column $j$.
This defines the placement of the pin uniquely.
For example: $r_2c_3r_2c_1r_1$
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2 c_3 r_2 c_1 r_1 c_2$

Example

![Diagram showing pin placement examples](image-url)
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2c_3r_2c_1r_1c_2r_1$
Letter $r_i$: place a pin in row $i$.
Letter $c_j$: place a pin in column $j$.
This defines the placement of the pin uniquely.
For example: $r_2 c_3 r_2 c_1 r_1 c_2 r_1 c_3$
Encoding Grid Pin Sequences

- Letter $r_i$: place a pin in row $i$.
- Letter $c_j$: place a pin in column $j$.
- This defines the placement of the pin uniquely.
- For example: $r_2 c_3 r_2 c_1 r_1 c_2 r_1 c_3 r_2$

Example

```
 r3
 o

 r2
   o

 r1
   o
```

```
 c1  c2  c3
 o o o
```