Pin-Permutations and Structure in Permutation Classes

Frédérique Bassino   Mathilde Bouvel   Dominique Rossin

Journées de Combinatoire de Bordeaux, feb. 2009

LIAFA

Main result of the talk

Conjecture[Brignall, Ruškuc, Vatter]:
The pin-permutation class has a rational generating function.

Theorem: The generating function of the pin-permutation class is

$$P(z) = z\frac{8z^6 - 20z^5 - 4z^4 + 12z^3 - 9z^2 + 6z - 1}{8z^8 - 20z^7 + 8z^6 + 12z^5 - 14z^4 + 26z^3 - 19z^2 + 8z - 1}$$

Technique for the proof:

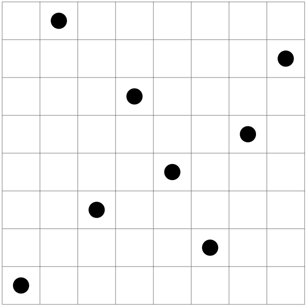
- Characterize the decomposition trees of pin-permutations
- Compute the generating function of *simple* pin-permutations
- Put things together to compute the generating function of pin-permutations

Outline of the talk

Representations of permutations

Permutation: Bijective map from $[1..n]$ to itself

- One-line representation:
  $\sigma = 1\ 8\ 3\ 6\ 4\ 2\ 5\ 7$
- Two-line representation:
  $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 8 & 3 & 6 & 4 & 2 & 5 & 7 \end{pmatrix}$
- Cyclic representation:
  $\sigma = (1)\ (2\ 8\ 7\ 5\ 4\ 6)\ (3)$
- Graphical representation:

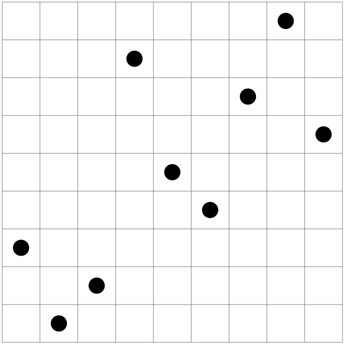


Patterns in permutations

Pattern relation $\preccurlyeq$:
$\pi \in S_k$ is a pattern of $\sigma \in S_n$ when $\exists\ 1 \leq i_1 < \ldots < i_k \leq n$ such that $\sigma_{i_1} \ldots \sigma_{i_k}$ is order-isomorphic to π.
We write $\pi \preccurlyeq \sigma$.

Equivalently: Normalizing $\sigma_{i_1} \ldots \sigma_{i_k}$ on $[1..k]$ yields π.

Example: $1\,2\,3\,4 \preccurlyeq 3\,\mathbf{1}\,\mathbf{2}\,8\,\mathbf{5}\,4\,\mathbf{7}\,9\,6$ since $1\,2\,5\,7 \equiv 1\,2\,3\,4$.

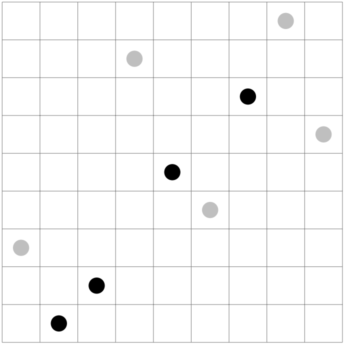


Patterns in permutations

Pattern relation $\preccurlyeq$:
$\pi \in S_k$ is a pattern of $\sigma \in S_n$ when $\exists\ 1 \leq i_1 < \ldots < i_k \leq n$ such that $\sigma_{i_1} \ldots \sigma_{i_k}$ is order-isomorphic to π.
We write $\pi \preccurlyeq \sigma$.

Equivalently: Normalizing $\sigma_{i_1} \ldots \sigma_{i_k}$ on $[1..k]$ yields π.

Example: $1\,2\,3\,4 \preccurlyeq 3\,\mathbf{1}\,\mathbf{2}\,8\,\mathbf{5}\,4\,\mathbf{7}\,9\,6$
since $1\,2\,5\,7 \equiv 1\,2\,3\,4$.



Classes of permutations

Class of permutations: set downward closed for $\preccurlyeq$
Equivalently: $\sigma \in \mathcal{C}$ and $\pi \preccurlyeq \sigma \Rightarrow \pi \in \mathcal{C}$

$S(B)$: the class of permutations avoiding all the patterns in the basis B.

Prop.: Every class $\mathcal{C}$ is characterized by its basis:

$$\mathcal{C} = S(B) \text{ for } B = \{\sigma \notin \mathcal{C} : \forall \pi \preccurlyeq \sigma \text{ with } \pi \neq \sigma, \pi \in \mathcal{C}\}$$

Basis may be finite or infinite.

Enumeration[Stanley-Wilf, Marcus-Tardos]: $|S_n(B)| \leq c_B^n$

Classes of permutations

Class of permutations: set downward closed for $\preccurlyeq$
Equivalently: $\sigma \in \mathcal{C}$ and $\pi \preccurlyeq \sigma \Rightarrow \pi \in \mathcal{C}$

$S(B)$: the class of permutations avoiding all the patterns in the basis B.

Prop.: Every class $\mathcal{C}$ is characterized by its basis:

$$\mathcal{C} = S(B) \text{ for } B = \{\sigma \notin \mathcal{C} : \forall \pi \preccurlyeq \sigma \text{ with } \pi \neq \sigma, \pi \in \mathcal{C}\}$$

Basis may be finite or infinite.

Enumeration[Stanley-Wilf, Marcus-Tardos]: $|S_n(B)| \leq c_B^n$

Studying classes of permutations

Pattern-avoidance point of view:
Definition by a basis of excluded patterns.

- Enumeration
- Exhaustive generation

Examples:

- $S(213, 312)$
- $S(4231)$
- $S(1\,2\ldots k)$

Structure in permutation classes:
Definition by a property stable for patterns.

- Characterization of the permutations
  - ↪ with excluded patterns
  - ↪ with a recursive description
- Properties of the generating function
- Algorithms for membership

Examples:

- Stack sortable $= S(231)$
- Separable $= S(2413, 3142)$
- Pin-permutations

Simple permutations

Interval = window of elements of σ whose values form a range
Example: 5 7 4 6 is an interval of 2 5 7 4 6 1 3

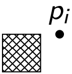
Simple permutation = has no interval except $1, 2, \ldots, n$ and σ
Example: 3 1 7 4 6 2 5 is simple. *Smallest ones*: 1 2, 2 1, 2 4 1 3, 3 1 4 2

Pin-permutations: used for deciding whether $\mathcal{C}$ contains finitely many simple permutations
Thm[Albert Atkinson]: $\mathcal{C}$ contains finitely many simple permutations
$\Rightarrow \mathcal{C}$ has an algebraic generating function

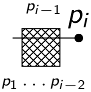
Decomposition trees: formalize the idea that simple permutations are "building blocks" for all permutations

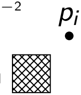
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

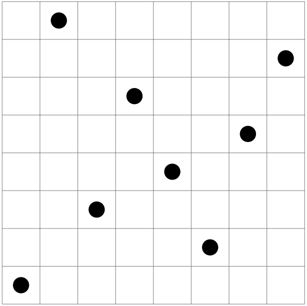
- and

• the separation condition 

• or the independence condition 

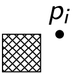
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

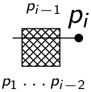


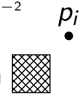
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

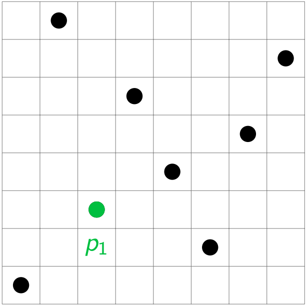
- and

- the separation condition 

- or the independence condition 

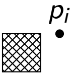
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

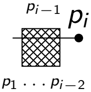


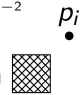
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

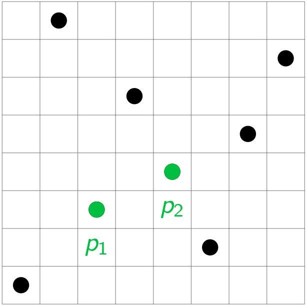
- and

- the separation condition 

- or the independence condition 

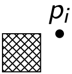
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

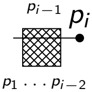


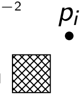
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

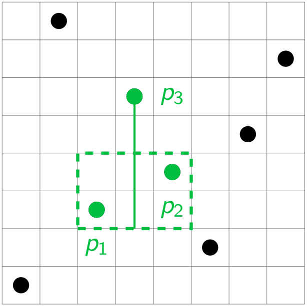
- and

• the separation condition 

• or the independence condition 

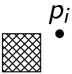
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

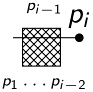
Example:

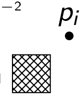


Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

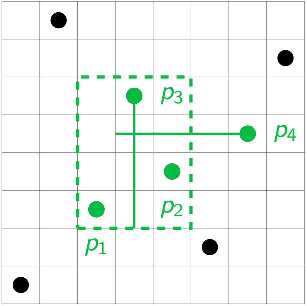
- the externality condition 

- and
  - the separation condition 

  - or the independence condition 

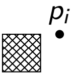
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

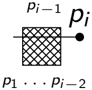


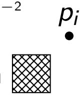
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

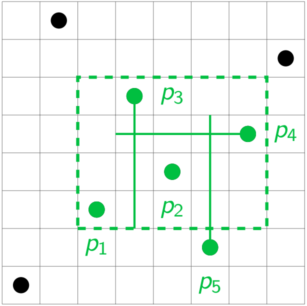
- and

• the separation condition 

• or the independence condition 

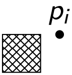
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

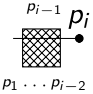


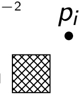
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

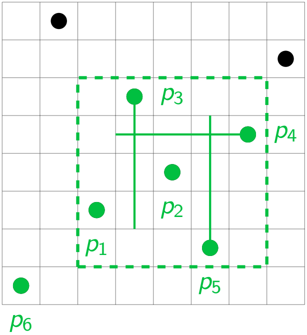
- and

- the separation condition 

- or the independence condition 

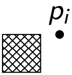
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

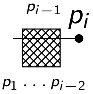


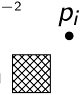
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

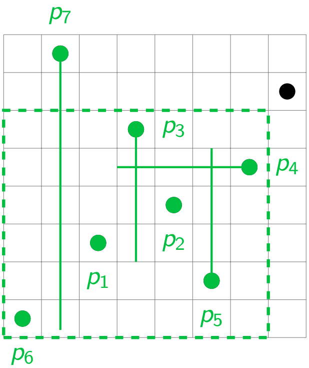
- and

- the separation condition 

- or the independence condition 

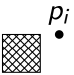
 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:

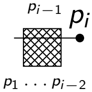


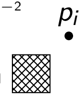
Pin representations

Pin representation of σ = sequence $(p_1, \ldots, p_n)$ such that each p_i satisfies

- the externality condition 

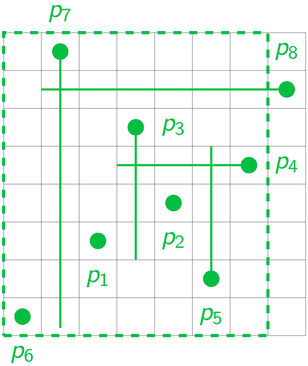
- and

• the separation condition 

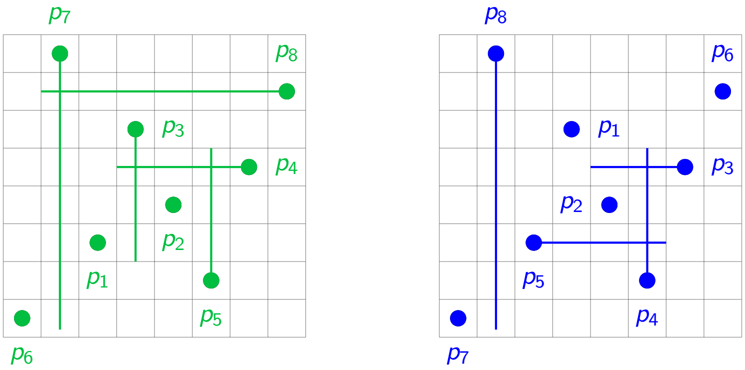
• or the independence condition 

 = *bounding box* of $\{p_1, \ldots, p_{i-1}\}$

Example:



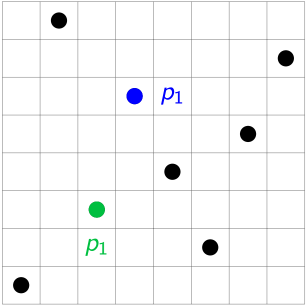
Non-uniqueness of pin representation



Active points

Active point of σ:
p_1 for some pin representation p of σ

Example:

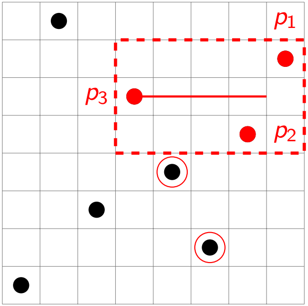


Active points

Active point of σ:
p_1 for some pin representation p of σ

Remark:
Not every point is an active point.

Example:

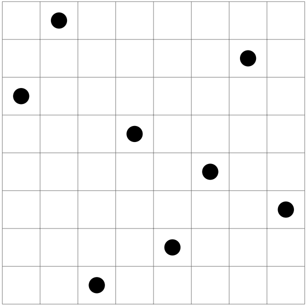


The class of pin-permutations

Fact: Not every permutation admits pin representations.

Def: Pin-permutation = that has a pin representation.

Example 1:



The class of pin-permutations

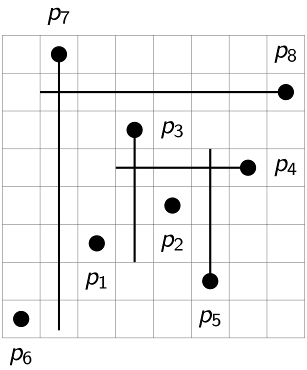
Fact: Not every permutation admits pin representations.

Def: Pin-permutation = that has a pin representation.

Thm: Pin-permutations are a permutation class.

Idea of the proof: σ has a pin representation $p \Rightarrow$ for $\tau \prec \sigma$ remove the same points in p.

Example 2:



The class of pin-permutations

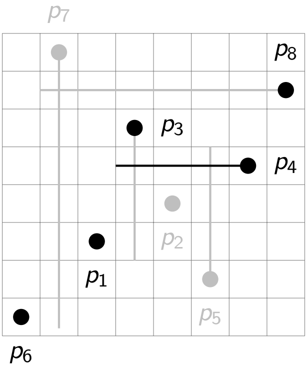
Fact: Not every permutation admits pin representations.

Def: Pin-permutation = that has a pin representation.

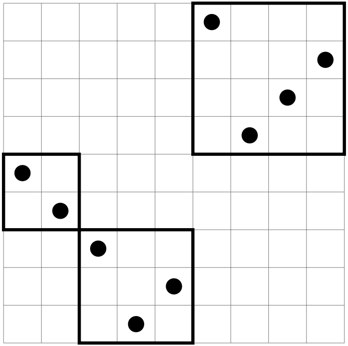
Thm: Pin-permutations are a permutation class.

Idea of the proof: σ has a pin representation $p \Rightarrow$ for $\tau \prec \sigma$ remove the same points in p.

Example 2:



Substitution decomposition



Definitions

Inflation:
$\pi[\alpha_1, \alpha_2, \ldots, \alpha_k]$

Example:
$213[21, 312, 4123] =$
$54\,312\,9678$

Substitution decomposition

Results

Prop.[Albert Atkinson]: $\forall \sigma, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \ldots, \alpha_k]$.
If $\pi = 12$ (21), for unicity, α_1 is *plus (minus) -indecomposable*.

Thm [Albert Atkinson]: (Wreath-closed) class $\mathcal{C}$ containing finitely many simple permutations $\Rightarrow$

- $\mathcal{C}$ is finitely based.
- $\mathcal{C}$ has an algebraic generating function.

Strong interval decomposition

Special case on permutations of the modular decomposition on graphs.

Thm: Every σ can be uniquely decomposed as

- $12 \dots k[\alpha_1, \dots, \alpha_k]$, with the α_i plus-indecomposable
- $k \dots 21[\alpha_1, \dots, \alpha_k]$, with the α_i minus-indecomposable
- $\pi[\alpha_1, \dots, \alpha_k]$, with π simple of size ≥ 4

Remarks:

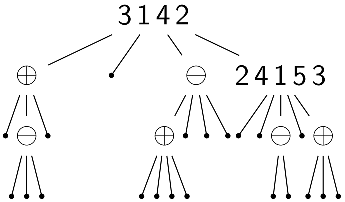
- This decomposition is unique without any further restriction.
- The α_i are the maximal strong intervals of σ.

Decompose the α_i recursively to get the decomposition tree.

Decomposition tree

Example: The substitution decomposition tree of $\sigma =$

10 13 12 11 14 1 18 19 20 21 17 16 15 4 8 3 2 9 5 6 7



Notations and properties:

- $\oplus = 12\ldots k$ and $\ominus = k\ldots 21$ = linear nodes.
- π simple of size ≥ 4 = prime nodes.
- No $\oplus - \oplus$ or $\ominus - \ominus$ egde.
- Decomposition trees of permutations are ordered.
- N.B.: Modular decomposition trees are unordered.

Bijection between decomposition trees and permutations.

On using decomposition trees

Algorithms:

- Computation in linear time
- Used in "efficient" algorithms for
  - ↪ Longest common pattern problem
  - ↪ Sorting by reversal
  - ↪ Computing perfect DCJ rearrangements

Examples in combinatorics: Use the bijective correspondance between decomposition trees and permutations.

- Wreath-closed classes: all trees on a given set of nodes
- Classes defined by a property: characterize the trees rather than the permutations
  - ↪ Separable permutations
  - ↪ Pin-permutations

Theorem

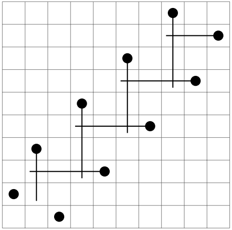
σ is a pin-permutation iff its decomposition tree satifies:

- Any linear node $\oplus$ ($\ominus$) has at most one child that is not an ascending (descending) weaving permutation
- For any prime node labelled by π, π is a simple pin-permutation and
  - all of its children are leaves
  - it has exactly one child that is not a leaf, and it inflates one active point of π
  - π is an ascending (descending) quasi-weaving permutation and exactly two children are not leaves
    - ↪ one is 12 (21) inflating the auxiliary substitution point of π
    - ↪ the other one inflates the main substitution point of π

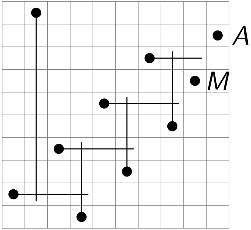
Definitions

Active point σ: there is a pin representation of σ starting with it.

Weaving permutation



Quasi-weaving permutation



Both are ascending. Other are obtained by symmetry.

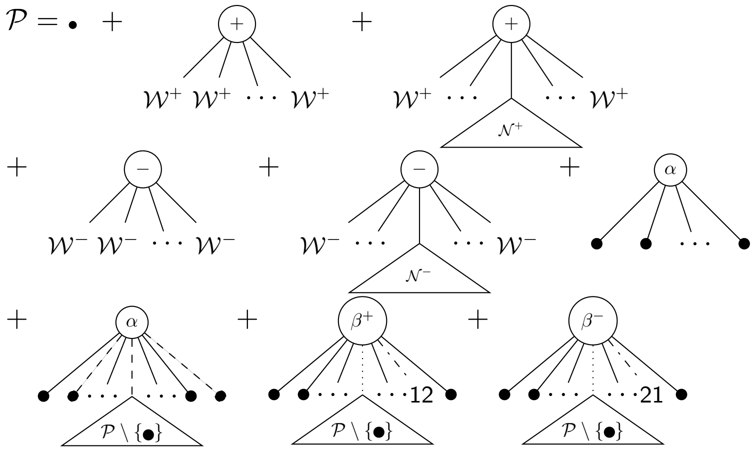
Enumeration: 4 $(= 2 + 2)$ weaving and 8 $(= 4 + 4)$ quasi-weaving permutations of size n, except for small n.

Theorem

σ is a pin-permutation iff its decomposition tree satifies:

- Any linear node $\oplus$ ($\ominus$) has at most one child that is not an ascending (descending) weaving permutation
- For any prime node labelled by π, π is a simple pin-permutation and
  - all of its children are leaves
  - it has exactly one child that is not a leaf, and it inflates one active point of π
  - π is an ascending (descending) quasi-weaving permutation and exactly two children are not leaves
    - $\hookrightarrow$ one is 12 (21) inflating the auxiliary substitution point of π
    - $\hookrightarrow$ the other one inflates the main substitution point of π

Theorem: more trees!



Basic generating functions involved

Weaving permutations: $W^+(z) = W^-(z) = W(z) = \frac{z+z^3}{1-z}$.
Remark: $\mathcal{W}^+ \cap \mathcal{W}^- = \{1, 2431, 3142\}$

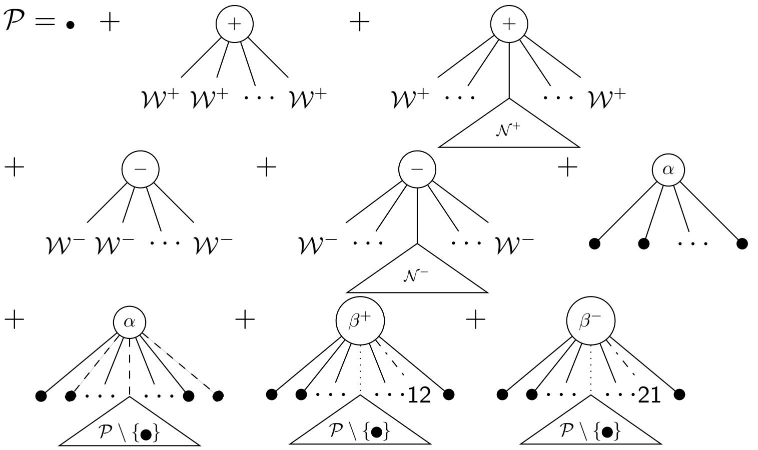
Quasi-weaving permutations:
$QW^+(z) = QW^-(z) = QW(z) = \frac{4z^4}{1-z}$.

Trees $\mathcal{N}^+$ and $\mathcal{N}^-$: pin-permutations except ascending (descending) weaving permutations and those whose root is $\oplus$ ($\ominus$).

$$N^+(z) = N^-(z) = N(z) = \frac{(z^3+2z-1)(z^3+P(z)z^3+2P(z)z+z-P(z))}{1-2z+z^2}$$

$P(z)$ = generating function of pin-permutations.

Theorem: more trees!



Generating functions of **simple** pin-permutations

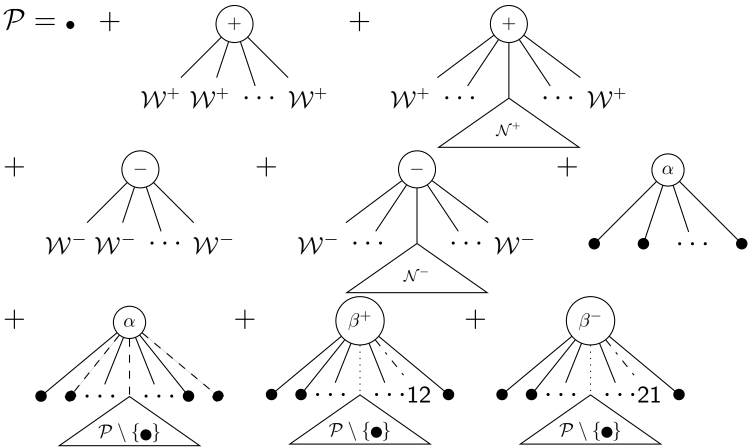
- Enumerate pin representations encoding simple pin-permutations.
- Characterize how many pin representations for a simple pin-permutation.
- Describe number of active points in simple pin-permutations.

Simple pin representations: $SiRep(z) = 8z^4 + \frac{32z^5}{1-2z} - \frac{16z^5}{1-z}$

Simple pin-permutations: $Si(z) = 2z^4 + 6z^5 + 32z^6 + \frac{128z^7}{1-2z} - \frac{28z^7}{1-z}$

Simple pin-permutations with multiplicity = number of active points: $SiMult(z) = 8z^4 + 26z^5 + 84z^6 + \frac{256z^7}{1-2z} - \frac{40z^7}{1-z}$

Theorem: more trees!



The rational generating function of pin-permutations

Equation on trees ⇒ equation on generating functions:

$$\begin{aligned}P(z) &= z + \frac{W^+(z)^2}{1-W^+(z)} + \frac{2W^+(z) - W^+(z)^2}{(1-W^+(z))^2} N^+(z)\\ &+ \frac{W^-(z)^2}{1-W^-(z)} + \frac{2W^-(z) - W^-(z)^2}{(1-W^-(z))^2} N^-(z) + Si(z)\\ &+ SiMult(z)\left(\frac{P(z)-z}{z}\right) + QW^+(z)\left(z\frac{P(z)-z}{z}\right) + QW^-(z)\left(z\frac{P(z)-z}{z}\right)\end{aligned}$$

Generating function of pin-permutations:

$$P(z) = z\frac{8z^6-20z^5-4z^4+12z^3-9z^2+6z-1}{8z^8-20z^7+8z^6+12z^5-14z^4+26z^3-19z^2+8z-1}$$

First terms: 1, 2, 6, 24, 120, 664, 3596, 19004, 99596, 521420, ...

Conclusion and open question

Overview of the results:

- Class of pin-permutations define by a graphical property
- Characterization of the associated decomposition trees
- Enumeration of simple pin-permutations

⇒ Generating function of the pin-permutation class

- Rationality of the generating function

Characterization of the pin-permutation class:

✓ by a recursive description

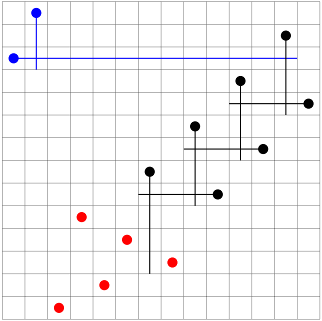
? by a (finite?) basis of excluded patterns

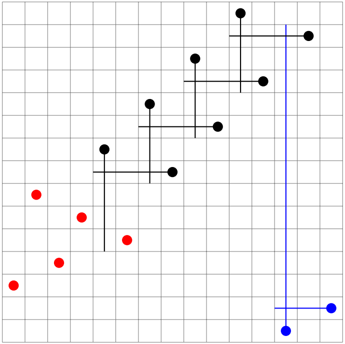
This basis is infinite, but yet unknown.

Infinite antichain in the basis

Prop. σ is in the basis $\Leftrightarrow$ σ is not a pin-permutation but any strict pattern of σ is.

We describe (σ_n) an infinite antichain in the basis:





Perspectives

Thm[Brignall et al.]: $\mathcal{C}$ a class given by its finite basis B. It is decidable whether $\mathcal{C}$ contains infinitely many simple permutations

Procedure: Check whether $\mathcal{C}$ contains arbitrarily long

- parallel alternations — Easy, Polynomial
- wedge simple permutations — Easy, Polynomial
- proper pin-permutations — Difficult, Complexity?

Analysis of the procedure for proper pin-permutations
$\Rightarrow$ Polynomial construction using automata techniques except last step *(Determinization of a transducer)*
$\Rightarrow$ makes the construction exponential

Better knowlegde of pin-permutations $\Rightarrow$ improve this complexity ?