

Automatic Enumeration and Random Generation for pattern-avoiding Permutation Classes

Adeline Pierrot

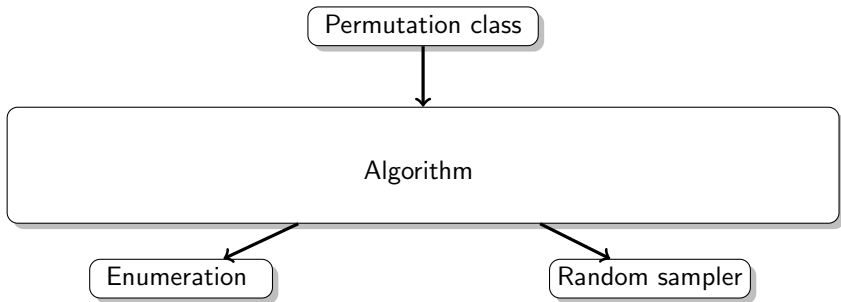
Institute of Discrete Mathematics and Geometry, TU Wien (Vienna)

Permutation Patterns 2014

Joint work with F. Bassino, M. Bouvel, C. Pivoteau and D. Rossin

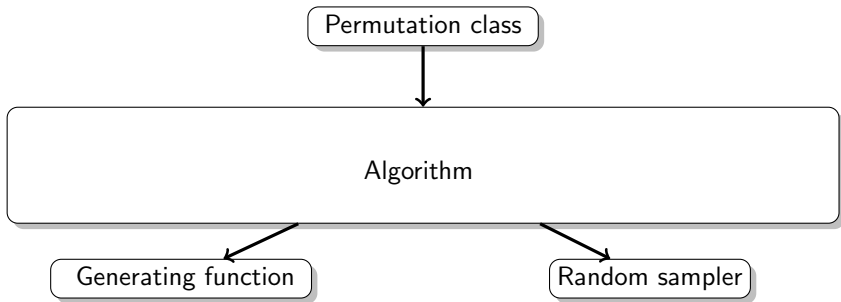
Main Result

Enumeration and Random Generation for Permutation Classes



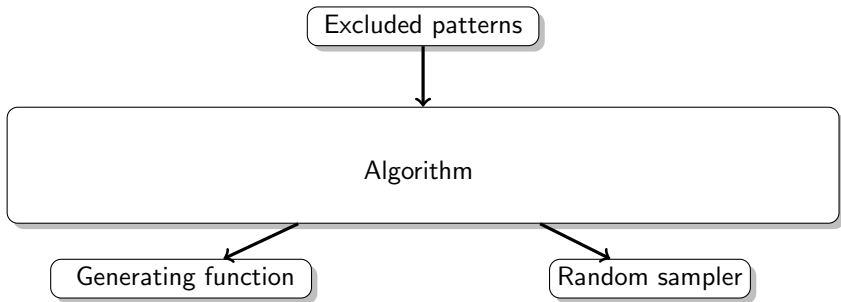
Main Result

Enumeration and Random Generation for Permutation Classes

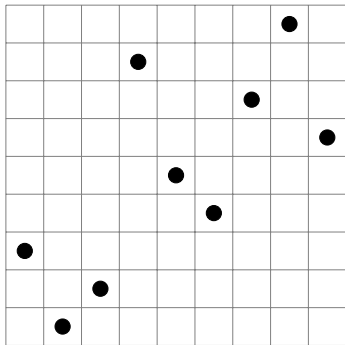


Main Result

Enumeration and Random Generation for Permutation Classes



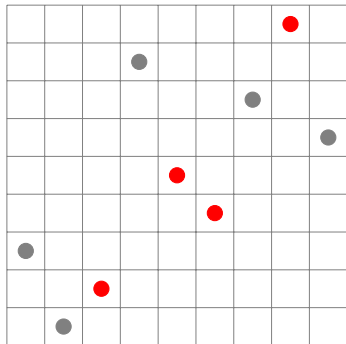
Patterns in permutations

Permutation : $\sigma = 312854796$ 

Patterns in permutations

Permutation : $\sigma = 312854796$

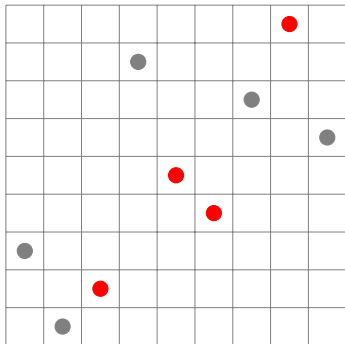
Pattern : $1324 \preceq 312854796$ since $2549 \equiv 1324$.



Patterns in permutations

Permutation : $\sigma = 312854796$

Pattern : $1324 \preceq 312854796$ since $2549 \equiv 1324$.

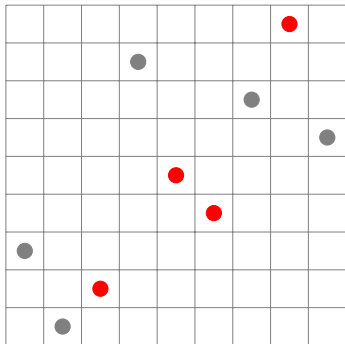


σ contains 1324
but avoids 4321

Patterns in permutations

Permutation : $\sigma = 312854796$

Pattern : $1324 \preceq 312854796$ since $2549 \equiv 1324$.



σ contains 1324
but avoids 4321

Remark : σ, π as input, deciding whether $\pi \preceq \sigma$ is NP-complete.

Permutation Classes

Class of permutations = set downward closed for $\preceq$:

$$\sigma \in \mathcal{C} \text{ and } \pi \preceq \sigma \Rightarrow \pi \in \mathcal{C}$$

Example : Increasing permutations $\cup_{n=1}^{\infty} \{12 \dots n\}$

$Av(B)$: the set of permutations avoiding all the elements of B .

Example : $Av(21) = \cup_{n=1}^{\infty} \{12 \dots n\}$

Permutation Classes

Class of permutations = set downward closed for $\preceq$:

$$\sigma \in \mathcal{C} \text{ and } \pi \preceq \sigma \Rightarrow \pi \in \mathcal{C}$$

Example : Increasing permutations $\cup_{n=1}^{\infty} \{12 \dots n\}$

$Av(B)$: the set of permutations avoiding all the elements of B .

Example : $Av(21) = \cup_{n=1}^{\infty} \{12 \dots n\}$

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

$$\forall \mathcal{C}, \exists \text{ a unique antichain } B \text{ s.t. } \mathcal{C} = Av(B)$$

B finite or infinite.

Two points of view :

$\mathcal{C}$ given by B / by a property stable for $\preceq$

Issues

Algorithms

- Pattern matching (NP-complete : 1998, FPT : 2014)
- Test the membership to a class

Combinatorics

- Above all enumeration
 - ↪ Lots of existing ad hoc results about a given class
- Search for general theory
 - ↪ Search structure in classes

Issues

Algorithms

- Pattern matching (NP-complete : 1998, FPT : 2014)
- Test the membership to a class

Combinatorics

- Above all enumeration
 - ↪ Lots of existing ad hoc results about a given class
- Search for general theory
 - ↪ Search structure in classes

Theorem[Albert, Atkinson, 2005] :

$\mathcal{C}$ contains finitely many simple permutations

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

Issues

Algorithms

- Pattern matching (NP-complete : 1998, FPT : 2014)
- Test the membership to a class

Combinatorics

- Above all enumeration
 - ↪ Lots of existing ad hoc results about a given class
- Search for general theory
 - ↪ Search structure in classes

Theorem[Albert, Atkinson, 2005] :

$\mathcal{C}$ contains finitely many simple permutations

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

This talk : Automatic process for enumeration / generation

Main tool : Recursive decomposition

Main tool : Substitution decomposition

Substitution decomposition : recursively describe discrete objects by decomposing them in **core items** (prime structures).

Existence and **unicity** of the decomposition.

Examples :

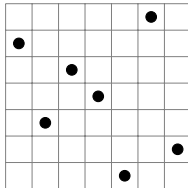
prime factorization of integers (core items = prime numbers),
modular decomposition of graphs (core items = prime graphs)...

Core items in the case of permutations : **simple permutations**.

Simple permutations

Block = window of elements of σ whose values form a range

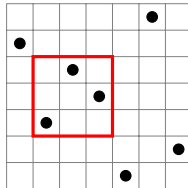
Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutations

Block = window of elements of σ whose values form a range

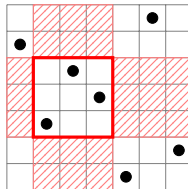
Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutations

Block = window of elements of σ whose values form a range

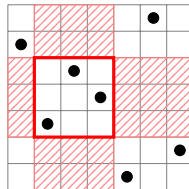
Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutations

Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2

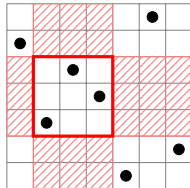


Simple permutation = has no block

Simple permutations

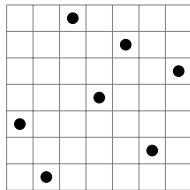
Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

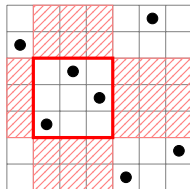
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.



Simple permutations

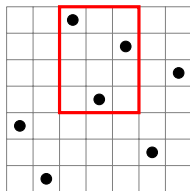
Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

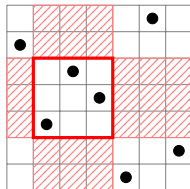
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.



Simple permutations

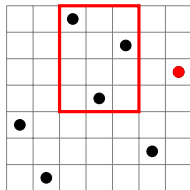
Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

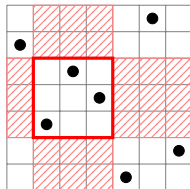
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.



Simple permutations

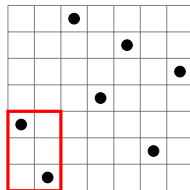
Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

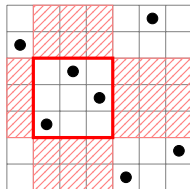
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.



Simple permutations

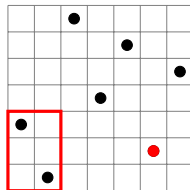
Block = window of elements of σ whose values form a range

Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

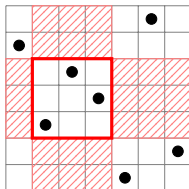
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.



Simple permutations

Block = window of elements of σ whose values form a range

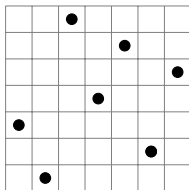
Example : 3 5 4 is a block of 6 **3 5 4** 1 7 2



Simple permutation = has no block

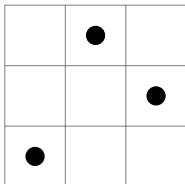
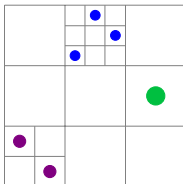
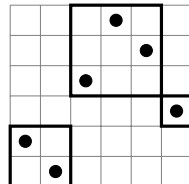
Example : 3 1 7 4 6 2 5 is simple,
6 3 5 4 1 7 2 is not simple.

Smallest ones : 1 2, 2 1, 2 4 1 3, 3 1 4 2



Substitution

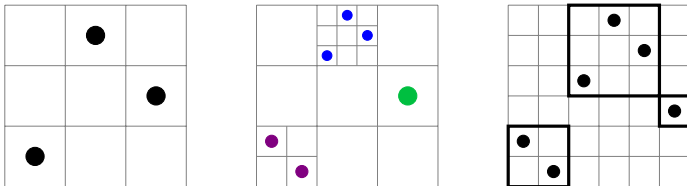
Substitution $\sigma[\pi^1, \dots, \pi^n]$: Replace each point σ_i by a block π^i .



Example : $1\ 3\ 2[2\ 1, 1\ 3\ 2, 1] = 2\ 1\ 4\ 6\ 5\ 3$.

Substitution

Substitution $\sigma[\pi^1, \dots, \pi^n]$: Replace each point σ_i by a block π^i .

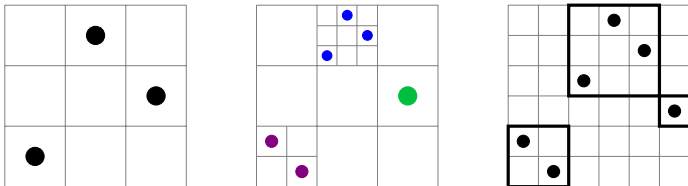


Example : $132[21, 132, 1] = 214653$.

Remark : $\sigma[\pi^1, \dots, \pi^n] \in \mathcal{C} \Rightarrow \sigma, \pi^1, \dots, \pi^n \in \mathcal{C}$

Substitution

Substitution $\sigma[\pi^1, \dots, \pi^n]$: Replace each point σ_i by a block π^i .



Example : $132[21, 132, 1] = 214653$.

Remark : $\sigma[\pi^1, \dots, \pi^n] \in \mathcal{C} \Rightarrow \sigma, \pi^1, \dots, \pi^n \in \mathcal{C}$

Substitution-closed class :

$\sigma, \pi^1, \dots, \pi^n \in \mathcal{C} \Rightarrow \sigma[\pi^1, \dots, \pi^n] \in \mathcal{C}$.

Substitution decomposition

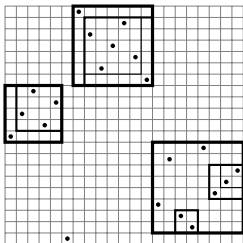
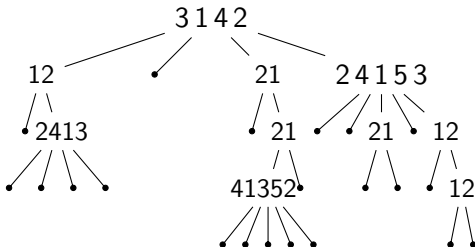
Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$.

Substitution decomposition

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$.
If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.

Substitution decomposition

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$.
 If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.



Example : $\sigma = 10\ 12\ 14\ 11\ 13\ 1\ 21\ 19\ 16\ 18\ 20\ 17\ 15\ 4\ 8\ 3\ 2\ 9\ 5\ 6\ 7$
 $= 3142 [13524, 1, 7524631, 37218456]$
 $= 3142 [12[1, 2413], 1, 21[1, 524631], 24153[1, 1, 21, 1, 123]]$
 $= \dots$

Decomposition $\Rightarrow$ Equation ?

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$
If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.

Consequence : For any permutation class $\mathcal{C}$,

$$\mathcal{C} \subset \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple } \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 12\text{-indecomposable}\}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 21\text{-indecomposable}\}$

Decomposition $\Rightarrow$ Equation ?

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$
If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.

Consequence : If $\mathcal{C}$ is a substitution-closed class (containing $12, 21$)

$$\mathcal{C} \subset \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple } \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 12\text{-indecomposable}\}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 21\text{-indecomposable}\}$

Decomposition $\Rightarrow$ Equation ?

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$
If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.

Consequence : If $\mathcal{C}$ is a substitution-closed class (containing $12, 21$)

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple } \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 12\text{-indecomposable}\}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 21\text{-indecomposable}\}$

Decomposition $\Rightarrow$ Equation ?

Theorem[Albert Atkinson 05] : $\forall \sigma \neq 1, \exists$ a unique simple permutation π and unique α_i such that $\sigma = \pi[\alpha_1, \dots, \alpha_k]$
If $\pi = 12$ (or 21), for unicity, α_1 is 12 (resp. 21) -indecomposable.

Consequence : If $\mathcal{C}$ is a substitution-closed class (containing $12, 21$)

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^+ = \{1\} \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^- = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple } \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 12\text{-indecomposable}\}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 21\text{-indecomposable}\}$

Combinatorial Specification (closed classes)

If $\mathcal{C}$ is a substitution-closed class (containing 12, 21)

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \bigsqcup_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^+ = \{1\} \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \bigsqcup_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^- = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus \bigsqcup_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

Combinatorial Specification (closed classes)

If $\mathcal{C}$ is a substitution-closed class (containing 12, 21)

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^+ = \{1\} \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^- = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

$\Leftrightarrow$ **Combinatorial Specification** if $\mathcal{S}_{\mathcal{C}}$ is finite and known
(Recursive description using combinatorial constructors)

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.
(see *Analytic Combinatorics* [Philippe Flajolet, Robert Sedgewick 2009])

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.
(see *Analytic Combinatorics* [Philippe Flajolet, Robert Sedgewick 2009])

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.
(see *Analytic Combinatorics* [Philippe Flajolet, Robert Sedgewick 2009])

$$\begin{aligned}\mathcal{C} &= \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}] \\ \rightarrow C(z) &= z + C^+(z).C(z) + C^-(z).C(z) + S_{\mathcal{C}}(C(z))\end{aligned}$$

Combinatorial Specification $\Rightarrow$ Equation

c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.
(see *Analytic Combinatorics* [Philippe Flajolet, Robert Sedgewick 2009])

$$\begin{aligned}\mathcal{C} &= \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}] \\ \rightarrow C(z) &= z + C^+(z).C(z) + C^-(z).C(z) + S_{\mathcal{C}}(C(z))\end{aligned}$$

$$\begin{array}{ll}\text{Same way} & C^+(z) = z + C^-(z).C(z) + S_{\mathcal{C}}(C(z)) \\ \text{and} & C^-(z) = z + C^+(z).C(z) + S_{\mathcal{C}}(C(z))\end{array}$$

Combinatorial Specification $\Rightarrow$ Equation

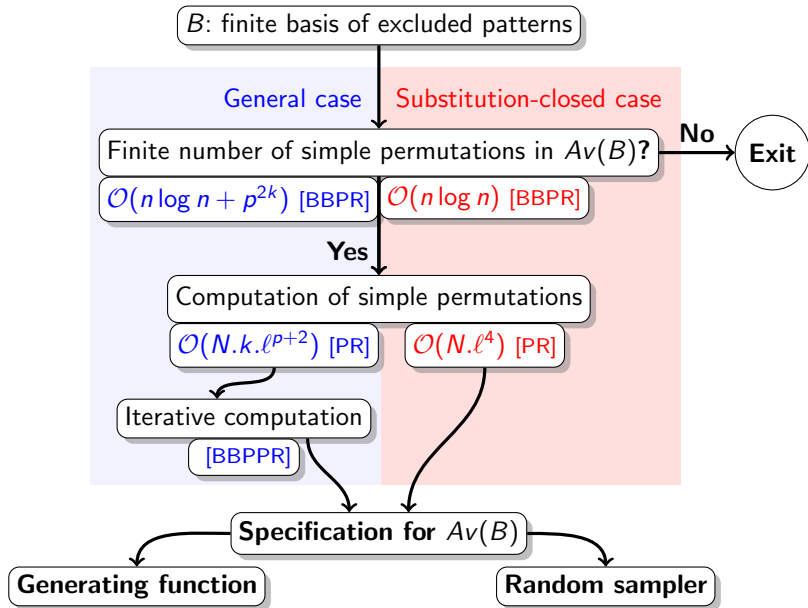
c_n = Number of permutations of size n in $\mathcal{C}$

Generating function $C(x) = \sum_{n=1}^{\infty} c_n \cdot x^n$

Symbolic method = a systematic translation mechanism :
combinatorial constructions $\rightarrow$ operations on generating functions.
(see *Analytic Combinatorics* [Philippe Flajolet, Robert Sedgewick 2009])

$$\begin{aligned}\mathcal{C} &= \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}] \\ \rightarrow C(z) &= z + C^+(z).C(z) + C^-(z).C(z) + S_{\mathcal{C}}(C(z))\end{aligned}$$

Combinatorial specification $\rightarrow$ **generating function** and **uniform random sampler** (recursive method or Boltzmann's method)



$$n = \sum_{\beta \in B} |\beta|, \quad k = |B|, \quad N = |\mathcal{S}_C|, \quad \ell = \max\{|\pi| : \pi \in \mathcal{S}_C\} \text{ et } p = \max\{|\beta| : \beta \in B\}$$

BBPPR = Bassino, Bouvel, Pierrot, Pivoteau, Rossin

Combinatorial Specification (closed classes)

If $\mathcal{C}$ is a substitution-closed class (containing 12, 21)

$$\mathcal{C} = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

$$\mathcal{C}^+ = \{1\} \uplus 21[\mathcal{C}^-, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

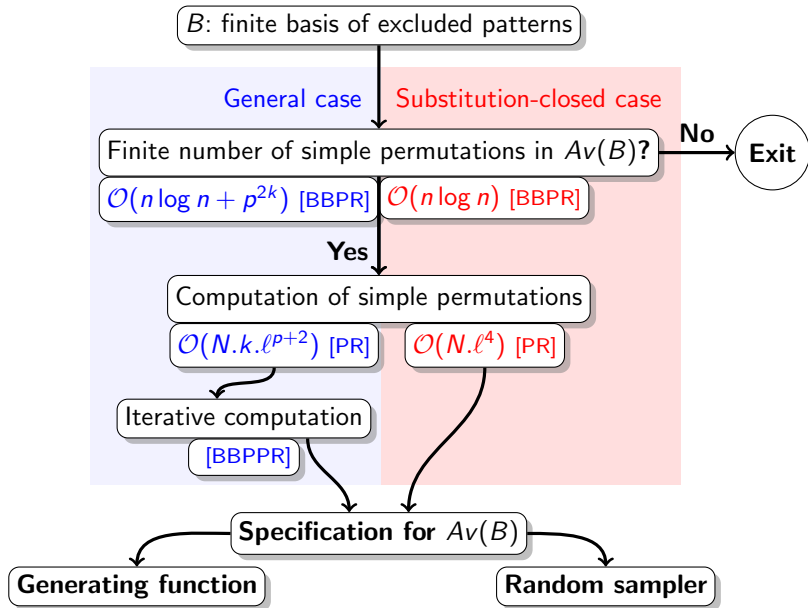
$$\mathcal{C}^- = \{1\} \uplus 12[\mathcal{C}^+, \mathcal{C}] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \pi[\mathcal{C} \dots \mathcal{C}]$$

Where $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

$\Leftrightarrow$ **Combinatorial Specification** if $\mathcal{S}_{\mathcal{C}}$ is finite and known
(Recursive description using combinatorial constructors)



$$n = \sum_{\beta \in B} |\beta|, \quad k = |B|, \quad N = |\mathcal{S}_C|, \quad \ell = \max\{|\pi| : \pi \in \mathcal{S}_C\} \text{ et } p = \max\{|\beta| : \beta \in B\}$$

BBPPR = Bassino, Bouvel, Pierrot, Pivoteau, Rossin

Finite number of simple permutations?

Thm[Brignall and al.] : Input = (finite) basis B . We can **decide** whether $\mathcal{C} = Av(B)$ contains finitely many simple permutations.

Procedure : Check if $\mathcal{C}$ contains finitely many

1. parallel alternations
2. wedge simple permutations
3. proper pin-permutations

Finite number of simple permutations ?

Thm[Brignall and al.] : Input = (finite) basis B . We can **decide** whether $\mathcal{C} = Av(B)$ contains finitely many simple permutations.

Procedure : Check if $\mathcal{C}$ contains finitely many

1. parallel alternations
2. wedge simple permutations
3. proper pin-permutations

Complexity : (with $n = \sum_{\pi \in B} |\pi|$, $k = |B|$ and $s \leq m = \max |\pi|$)

1. and 2. : pattern matching for patterns of size $\leq 4 \rightarrow \mathcal{O}(n \log n)$
3. : use words and automata theory
Brignall's procedure : $\mathcal{O}(n \cdot 8^m + 2^{k \cdot s \cdot 2^s})$
Our algorithm [BBPR] : $\mathcal{O}(n + s^{2k})$
Substitution-closed class : $\mathcal{O}(n)$.

Compute simple permutations

Prop[Schmerl, Trotter 93] : Any simple permutation of size n has a simple pattern of size $n - 1$ or $n - 2$.

Compute simple permutations

Prop[Schmerl, Trotter 93] : Any simple permutation of size n has a simple pattern of size $n - 1$ or $n - 2$.

Naive algorithm : Generate all simple permutations σ of size n and test if $\sigma \in \mathcal{C}$ until n s.t. no simple of size n or $n - 1$ belongs to $\mathcal{C}$.

Compute simple permutations

Prop[Schmerl, Trotter 93] : Any simple permutation of size n has a simple pattern of size $n - 1$ or $n - 2$.

Naive algorithm : Generate all simple permutations σ of size n and test if $\sigma \in \mathcal{C}$ until n s.t. no simple of size n or $n - 1$ belongs to $\mathcal{C}$.

Improvement[PR] : Restrict the number of tests :
Build candidates from simples of size $n - 1$ of $\mathcal{C}$
 $\longrightarrow < n.c^n$ tests instead of $n!$.

Compute simple permutations

Prop[Schmerl, Trotter 93] : Any simple permutation of size n has a simple pattern of size $n - 1$ or $n - 2$.

Naive algorithm : Generate all simple permutations σ of size n and test if $\sigma \in \mathcal{C}$ until n s.t. no simple of size n or $n - 1$ belongs to $\mathcal{C}$.

Improvement[PR] : Restrict the number of tests :
Build candidates from simples of size $n - 1$ of $\mathcal{C}$
 $\longrightarrow < n.c^n$ tests instead of $n!$.

Thm[PR] : σ, π simple, $\pi \prec \sigma \Rightarrow \exists \tau$ simple s.t. $\pi \preceq \underbrace{\tau \prec \sigma}_1$

Compute simple permutations

Prop[Schmerl, Trotter 93] : Any simple permutation of size n has a simple pattern of size $n - 1$ or $n - 2$.

Naive algorithm : Generate all simple permutations σ of size n and test if $\sigma \in \mathcal{C}$ until n s.t. no simple of size n or $n - 1$ belongs to $\mathcal{C}$.

Improvement[PR] : Restrict the number of tests :
Build candidates from simples of size $n - 1$ of $\mathcal{C}$
 $\longrightarrow < n.c^n$ tests instead of $n!$.

Thm[PR] : σ, π simple, $\pi \prec \sigma \Rightarrow \exists \tau$ simple s.t. $\pi \preceq \underbrace{\tau \prec \sigma}_1$

Improvement[PR] : If $\mathcal{C}$ substitution-closed, no need of pattern matching : just check permutations obtained by deleting a point to the candidate.

Combinatorial Specification

Recall : If $\mathcal{C}$ substitution-closed (and $12, 21 \in \mathcal{C}$)

$$\mathcal{C} = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \mathcal{C}^+ \quad \mathcal{C} \end{array} \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \mathcal{C}^- \quad \mathcal{C} \end{array} \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \mathcal{C} \quad \cdots \quad \mathcal{C} \end{array}$$

With $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable}\}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable}\}$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}} = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}} \quad \cdots \quad \hat{\mathcal{C}} \end{array}$$

With $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}} = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}} \quad \cdots \quad \hat{\mathcal{C}} \end{array}$$

With $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

$$\mathcal{C} = \hat{\mathcal{C}} \cap Av(B) = \hat{\mathcal{C}} \langle B \rangle$$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}}\langle B \rangle = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}} \quad \cdots \quad \hat{\mathcal{C}} \end{array} \langle B \rangle$$

With $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 12-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is 21-indecomposable} \}$

$$\mathcal{C} = \hat{\mathcal{C}} \cap Av(B) = \hat{\mathcal{C}}\langle B \rangle$$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}}\langle B \rangle = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}}\langle B_1 \rangle \cdots \hat{\mathcal{C}}\langle B_k \rangle \end{array}$$

With $\mathcal{S}_{\mathcal{C}} = \{\pi \in \mathcal{C} \mid \pi \text{ simple} \neq 12, 21\}$

$\mathcal{C}^+ = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 12\text{-indecomposable} \}$

$\mathcal{C}^- = \{\alpha \in \mathcal{C} \mid \alpha \text{ is } 21\text{-indecomposable} \}$

$$\mathcal{C} = \hat{\mathcal{C}} \cap Av(B) = \hat{\mathcal{C}}\langle B \rangle$$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}}\langle B \rangle = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}}\langle B_1 \rangle \cdots \hat{\mathcal{C}}\langle B_k \rangle \end{array}$$

Constraint Propagation $\rightarrow$ System of equations like :

$$\mathcal{C}_1 = 1 \uplus 12[\mathcal{C}_2, \mathcal{C}_3] \uplus 21[\mathcal{C}_4, \mathcal{C}_5] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \pi[\mathcal{C}_6, \dots, \mathcal{C}_k]$$

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

$$\hat{\mathcal{C}}\langle B \rangle = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}}\langle B_1 \rangle \cdots \hat{\mathcal{C}}\langle B_k \rangle \end{array}$$

Constraint Propagation $\rightarrow$ System of equations like :

$$\mathcal{C}_1 = 1 \uplus 12[\mathcal{C}_2, \mathcal{C}_3] \uplus 21[\mathcal{C}_4, \mathcal{C}_5] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \pi[\mathcal{C}_6, \dots, \mathcal{C}_k]$$

Iterative computation $\rightarrow$ Ambiguous system.

Combinatorial Specification

Consequence : If $\mathcal{C}$ not closed of closure $\hat{\mathcal{C}}$ (and $12, 21 \in \mathcal{C}$)

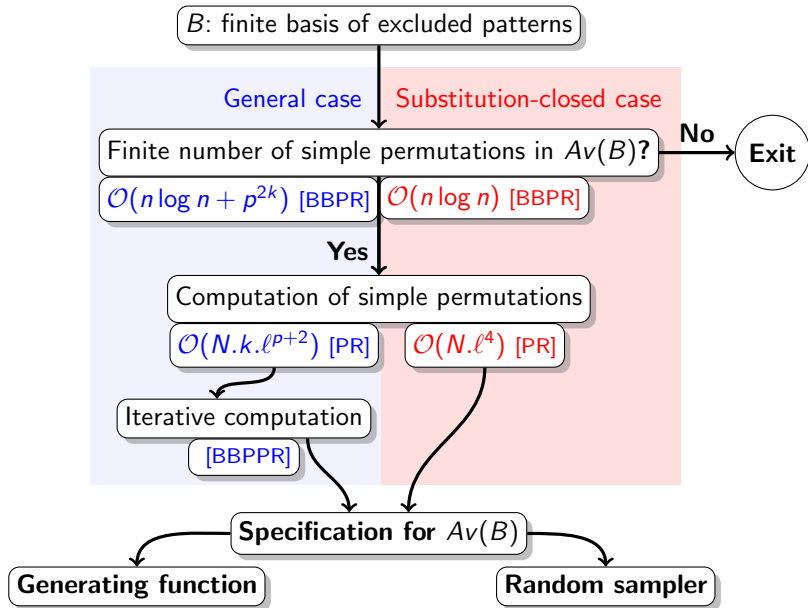
$$\hat{\mathcal{C}}\langle B \rangle = 1 \uplus \begin{array}{c} 12 \\ / \quad \backslash \\ \hat{\mathcal{C}}^+ \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \begin{array}{c} 21 \\ / \quad \backslash \\ \hat{\mathcal{C}}^- \quad \hat{\mathcal{C}} \end{array} \langle B \rangle \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \begin{array}{c} \pi \\ / \quad | \quad \backslash \\ \hat{\mathcal{C}}\langle B_1 \rangle \cdots \hat{\mathcal{C}}\langle B_k \rangle \end{array}$$

Constraint Propagation $\rightarrow$ System of equations like :

$$\mathcal{C}_1 = 1 \uplus 12[\mathcal{C}_2, \mathcal{C}_3] \uplus 21[\mathcal{C}_4, \mathcal{C}_5] \uplus \biguplus_{\pi \in \mathcal{S}_{\mathcal{C}}} \cup \pi[\mathcal{C}_6, \dots, \mathcal{C}_k]$$

Iterative computation $\rightarrow$ Ambiguous system.

$\rightarrow$ Disambiguation [AA05] inclusion-exclusion
 [BHV08] “query-complete sets”
 [BBPPR] mandatory patterns
 $A \cup B = A \cap \bar{B} \uplus \bar{A} \cap B \uplus A \cap B$



$$n = \sum_{\beta \in B} |\beta|, \quad k = |B|, \quad N = |\mathcal{S}_C|, \quad \ell = \max\{|\pi| : \pi \in \mathcal{S}_C\} \text{ et } p = \max\{|\beta| : \beta \in B\}$$

BBPPR = Bassino, Bouvel, Pierrot, Pivoteau, Rossin

Perspectives

- [Maple Library](#) for the permutation patterns community

Perspectives

- [Maple Library](#) for the permutation patterns community
- Random generation → [tests](#)

Perspectives

- [Maple Library](#) for the permutation patterns community
- Random generation $\rightarrow$ [tests](#)
- [Limit shape](#) of permutations ?

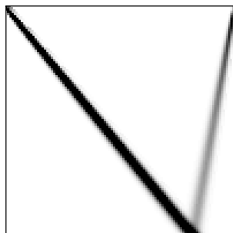


FIGURE: 30 000 random permutations of size 500 of $Av(2413, 1243, 2341, 531642, 41352)$.

Perspectives

- [Maple Library](#) for the permutation patterns community
- Random generation $\rightarrow$ [tests](#)
- [Limit shape](#) of permutations?
- Study the specifications obtained $\rightarrow$ [asymptotics](#)?

Perspectives

- **Maple Library** for the permutation patterns community
- Random generation → **tests**
- **Limit shape** of permutations?
- Study the specifications obtained → **asymptotics**?
- Finite number of simples : **Representative** behaviour of all permutation classes?

Perspectives

- **Maple Library** for the permutation patterns community
- Random generation → **tests**
- **Limit shape** of permutations?
- Study the specifications obtained → **asymptotics**?
- Finite number of simples : **Representative** behaviour of all permutation classes?
- **Generalization** (infinite number of simple permutations...)

Perspectives

- **Maple Library** for the permutation patterns community
- Random generation → **tests**
- **Limit shape** of permutations?
- Study the specifications obtained → **asymptotics**?
- Finite number of simples : **Representative** behaviour of all permutation classes?
- **Generalization** (infinite number of simple permutations...)

Thank you !