AUTOMATIC DISCOVERY OF STRUCTURAL RULES OF PERMUTATION CLASSES

CHRISTIAN BEAN, BJARKI AGUST GUDMUNDSSON, AND HENNING ULFARSSON

Abstract. We introduce an algorithm that conjectures the structure of a permutation class in the form of a disjoint cover of "rules"; similar to generalized grid classes. The cover is usually easily verified by a human and translated into an enumeration. The algorithm is successful on different inputs than other algorithms and can succeed with any polynomial permutation class. We apply it to every non-polynomial permutation class avoiding a set of length four patterns. The structures found by the algorithm can sometimes allow an enumeration of the permutation class with respect to permutation statistics, as well as choosing a permutation uniformly at random from the permutation class. We sketch a new algorithm formalizing the human verification of the conjectured covers.

1. Introduction

There has been a movement towards a generalized approach for enumerating permutation classes, which are sets of permutations defined by the avoidance of permutation patterns[1].

One of the first completely automatic approaches was enumeration schemes introduced by Zeilberger [1998] and considerably extended by Vatter [2008]. The goal of enumeration schemes is to break up a permutation class into smaller parts and find recurrence relations. There is no general theory for when a permutation class has an enumeration scheme.

The insertion encoding is an encoding of finite permutations, introduced by Albert et al. [2005]. It encodes how a permutation is built up by iteratively adding a new maximum element. In particular, they studied the permutation classes whose insertion encodings are regular languages, including giving a characterization of these permutation classes. For regular insertion encodable permutation classes, Vatter [2012] provides an algorithm for automatically computing the rational generating function.

Albert and Atkinson [2005] introduced the enumeration of a set of permutations by inflating the simple permutations in the set. A basis is not required but the set under inspection must contain only finitely many simple permutations. Checking whether this condition is satisfied is in general exponentially hard, but for wreath-closed permutation classes an $O(n \log n)$ algorithm exists, Bassino et al. [2010]. The whole procedure has not been implemented.

In the case of polynomial permutation classes[2], it was shown by Homberger and Vatter [2016], by combining results by Albert et al. [2013] and Huczynska and Vatter [2006], that each permutation class can be represented by a finite set of peg permutations[3]. From this finite set of peg

[1]We define these precisely at the end of the introduction.

[2]A permutation class $\mathcal{C}$ is said to be *polynomial* if the number of length n permutations, $|\mathcal{C}_n|$, is given by a polynomial for all sufficiently large n.

[3]We discuss peg permutations in Section 5

permutations, Homberger and Vatter [2016] give an automatic method to enumerate polynomial permutation classes. However, it is not known in general how to find the set of peg permutations.

We introduce Struct, an algorithm which takes as input a set of permutations or a permutation property and conjectures certain structural rules that can lead to a generating function enumerating the input. We focus specifically on permutation classes and assume a finite basis is known. We emphasize if the algorithms mention above succeed, then they output a proof, whereas our algorithm only outputs conjectures. We note however that the produced conjectures usually turn out to be easily verified by a human.

As a first example, consider the set $\mathcal{A}$ of permutations avoiding 231, which are the stack-sortable permutations, first considered by Knuth [1998]. It is well known that a non-empty permutation in $\mathcal{A}$ can be written as $\alpha n \beta$ where n is the largest element in the permutation, α and β avoid 231, and every element in α is smaller than every element in β. We can state the structure as a disjoint union of the two rules in Figure 1. The meaning of this representation is that a permutation is either

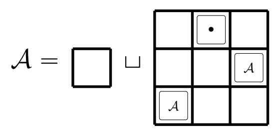


FIGURE 1. The structure of Av(231).

empty, and represented by the empty rule, or, it is non-empty and represented by the 3×3 rule. That rule is a recipe that says: given two permutations in $\mathcal{A}$, we can generate a longer permutation in $\mathcal{A}$ by placing the two permutations in the cells marked with $\mathcal{A}$ and placing a new largest element between them, e.g. 213 and 1234 will generate the permutation 21384567. This leads immediately to the well-known functional equation for the Catalan generating function, $A(x) = 1 + xA(x)^2$.

For the remainder of the introduction, we give the definitions used throughout. In Section 2 we discuss precisely how the algorithm works. In Sections 3 and 4 we apply Struct to specific permutation classes. In Section 5 we compare Struct to the other existing approaches mentioned above and show it will succeed for all polynomial permutation classes. In Section 6 we apply it to the bases $B \subseteq \mathcal{S}_4$. In Section 7 we consider improvements of the algorithm, relating to permutation statistics, uniform random choice from a permutation class, and how the conjectures can be turned automatically into theorems.

A *permutation* of length n is a word on $[n] = \{1, 2, \ldots, n\}$ that contains each letter exactly once. Let $\mathcal{S}_n$ be the set of all permutations of length n and $\mathcal{S} = \bigcup_{n \geqslant 0} \mathcal{S}_n$. Two words of the same length, $w = w_1 w_2 \ldots w_k$ and $v = v_1 v_2 \ldots v_k$, are *order isomorphic* if for all i and j, $w_i < w_j$ if and only if $v_i < v_j$. A permutation p is *contained* in a permutation π, denoted $p \preceq \pi$, if there is a subword in π that is order isomorphic to p. If π does not contain p then we say π *avoids* p. In this context, p is referred to as a (*classical permutation*) *pattern*.

Define the set $\Delta(\pi)$ to be the set of all patterns contained in π,

$$\Delta(\pi) = \{p \in \mathcal{S} : p \preceq \pi\}.$$

For a set of patterns P, let $\Delta(P) = \bigcup_{\pi \in P} \Delta(\pi)$. A *permutation class* is a set $\mathcal{C}$ such that $\Delta(\mathcal{C})$ equals $\mathcal{C}$. For a set of patterns P, the set

$$\mathrm{Av}(P) = \{\pi \in \mathcal{S} : \pi \text{ avoids all } p \in P\}$$

is a permutation class. Given a permutation class $\mathcal{C} = \text{Av}(B)$ we say that a set of patterns B is a *basis* of $\mathcal{C}$ if $\Delta(p) \cap B = \{p\}$ for each $p \in B$. It is easy to see that every permutation class has a (possibly infinite) basis. If $|B| = 1$ then we call $\text{Av}(B)$ a *principal* permutation class.

For a permutation class $\mathcal{C} = \text{Av}(P)$ we denote the set of all length n permutations in $\mathcal{C}$ as $\mathcal{C}_n$ or $\text{Av}_n(P)$. We also let $\mathcal{C}_{\leqslant n}$ and $\text{Av}_{\leqslant n}(P)$ be the set of permutations of length at most n in the permutation class. Two sets of patterns P and P' are *Wilf-equivalent* if $|\text{Av}_n(P)| = |\text{Av}_n(P')|$ for all n.

2. The algorithm

Before describing how the algorithm works we give a few more definitions. A *block* of a permutation class $\mathcal{C} = \text{Av}(B)$ is a permutation class $\mathcal{C}' = \text{Av}(B')$, containing infinitely many permutations, such that $B' \subseteq \Delta(B)$ and $B' \cap \Delta(p) \neq \varnothing$ for all p in B. We also allow the finite permutation class $\text{Av}(1) = \{\epsilon\}$ as a block for any permutation class $\mathcal{C}$. Additionally, if $1 \in \mathcal{C}$ we allow $\{1\}$ as a block, even though this is not a permutation class, and contains only one permutation. We call 1 "the point" and denote it with $\bullet$.

The *block set*, $\mathbb{M}(B)$, of $\mathcal{C}$ is the set of all blocks of $\mathcal{C}$. We note that for a finite basis B the block set of the permutation class $\text{Av}(B)$ is always finite. For example,

$$\mathbb{M}(\{231\}) = \{\text{Av}(1), \{\bullet\}, \text{Av}(12), \text{Av}(21), \text{Av}(231)\}.$$

Note that $\text{Av}(12, 21)$ is not a block since it is finite.

Before we assemble blocks into "rules" we review generalized grid classes introduced by Vatter [2011]. Given a permutation π of length n, and two subsets $X, Y \subseteq [n]$, then $\pi(X \times Y)$ is the permutation that is order isomorphic to the subword with indices from X and values in Y. For example $35216748([3,7] \times [2,6]) = 132$, from the subword 264.

Suppose M is a $t \times u$ matrix (indexed from left to right and bottom to top) whose entries are permutation classes. An *M-gridding* of a permutation π of length n is a pair of sequences $1 = c_1 \leqslant \cdots \leqslant c_{t+1} = n+1$ and $1 = r_1 \leqslant \cdots \leqslant r_{u+1} = n+1$ such that $\pi([c_k, c_{k+1}) \times [r_l, r_{l+1}))$ is in $M_{k,l}$ for all k in $[t]$ and ℓ in $[u]$. The *generalized grid class* of M, $\text{Grid}(M)$, consists of all permutations with an M-gridding.

A (*Struct*) *rule* $\mathcal{R}$ is a matrix whose entries are permutation classes, or the point $\bullet$, with the requirement that each permutation π in the grid class $\text{Grid}(\mathcal{R})$ has a unique $\mathcal{R}$-gridding. From now on we will abuse notation and use $\mathcal{R}$ to denote both the Struct rule and its grid class. Let $\mathcal{R}_{\leqslant n}$ be the set of permutations in $\mathcal{R}$ of length at most n.

For a given basis B, whose longest pattern is from S_ℓ, Struct tries to find the structure of the permutation class $\mathcal{C} = \text{Av}(B)$ in terms of rules. Using the default settings, the algorithm consists of four main steps

(1) Generate the block set[4] of the given basis B.
(2) Generate Struct rules[5], $\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_K$ up to size $(\ell+1) \times (\ell+1)$ with entries from the block set, satisfying $\mathcal{R}_{\leqslant \ell+2} \subseteq \text{Av}(B)$. The trivial rule $\boxed{\mathcal{C}}$ is discarded.

[4]Candidates for blocks are experimentally checked for being infinite: If a block is non-empty up to length $\ell+2$ it is kept.

[5]Candidates for Struct rules are experimentally checked to not create duplicates up to length $\ell+2$.

(3) Try to find a *cover* of $\mathrm{Av}_{\leqslant \ell+2}(B)$ with the rules from the previous step, i.e., write this set as a disjoint union

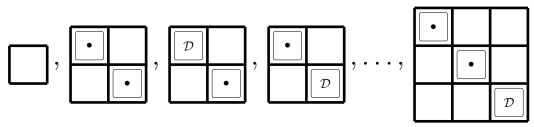
$$\mathrm{Av}_{\leqslant \ell+2}(B) = \mathcal{R}_{i_1, \leqslant \ell+2} \sqcup \cdots \sqcup \mathcal{R}_{i_k, \leqslant \ell+2}.$$

(4) If a cover is found, verify that it remains valid in length $\leqslant \ell + 4$, i.e.,

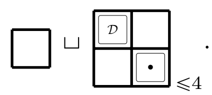
$$\mathrm{Av}_{\leqslant \ell+4}(B) = \mathcal{R}_{i_1, \leqslant \ell+4} \sqcup \cdots \sqcup \mathcal{R}_{i_k, \leqslant \ell+4}.$$

By providing different settings, the user can make the algorithm look for a cover with larger rules, or verify a found cover for longer permutations.

As a simple example consider the decreasing permutations $\mathcal{D} = \mathrm{Av}(12)$, where $\ell = 2$. Step (1) finds the block set $\mathbb{M}(\{12\}) = \{\mathrm{Av}(1), \{\bullet\}, \mathrm{Av}(12)\}$. Step (2) generates the rules



One of the covers found in step (3) is

$\mathcal{D}_{\leqslant 4} =$ 

Step (4) verifies this cover up to length 6. At this stage, a human must step in and verify that the cover remains valid for all lengths. In this case, it is obvious, and if $D(x)$ is the generating function then we can see from the cover that $D(x) = 1 + xD(x)$, so $D(x) = \frac{1}{1-x}$.

If the block set is large then the space of possible rules to consider in step (2) will be too large to exhaustively search for valid rules. To prune the search space we first check which blocks can share a row, column or diagonal without creating a pattern from the basis. This is used to recursively build the candidate Struct rules without considering every possibility. We also arrange the blocks in a poset where the relation is set containment. This is also used when creating the Struct rules; if a rule with a block A in a certain cell produces the same permutation twice, or a permutation outside of $\mathrm{Av}(B)$ then replacing A with a block $A' \supset A$ will also not work. Finally, in step (3) we use the integer linear programming solver Gurobi (Gurobi Optimization [2016]) to find a minimal cover if one exists; resorting to the SAT solver Lingeling (Biere [2013]) if Gurobi runs out of memory.

The implementation of Struct can be found on GitHub, Bean et al. [2017]. The conjectures that have been discovered can be found on the Permutation Pattern Avoidance Library (or PermPAL for short), Arnarson et al. [2017], alongside the conjectured enumerations.

3. Patterns of length three

As we saw in the previous section the decreasing permutations, $\mathcal{D} = \mathrm{Av}(12)$, have a cover from which we can read the generating function, $D(x) = \frac{1}{1-x}$. By symmetry the increasing permutations $\mathcal{I} = \mathrm{Av}(21)$ have the same generating function, which we denote $I(x)$. Throughout we will use the font $\mathcal{C}$ to denote a permutation class and $C(x)$ to denote its corresponding generating function.

In this section, we consider selected bases $B \subseteq \mathcal{S}_3$. All of the results in this section are known, see e.g., Simion and Schmidt [1985]. We include them here as we will need to reference them for permutation classes considered later. For the bases not mentioned in this section there exists a cover using at most 3×3 rules, with the exception of $\{123\}$, which will be treated in Section 7. Our approach has the appeal of being automatic: Every cover is conjectured by Struct and easy for a human to prove. Furthermore, reading the generating function from the cover is a routine exercise. Where possible the OEIS entry is referenced, Sloane [2017].

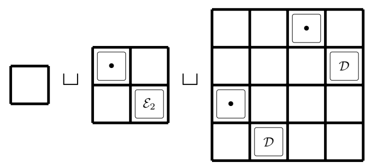
We already have from Figure 1 that $\mathcal{A} = \mathrm{Av}(231)$ has a cover that leads to the Catalan generating function, that is $A(x) = \frac{1-\sqrt{1-4x}}{2x}$.

Two patterns. We start by considering two patterns of length three.

Proposition 1 (Simion and Schmidt [1985], Proposition 11)**.** *The structure of the permutation class* $\mathcal{E}_2 = \mathrm{Av}(123, 231)$ *is given by the cover in Equation* (1)*. The enumeration is given by A152947 and the generating function is*

$$E_2(x) = \frac{1-2x+2x^2}{(1-x)^3} = 1 + x + 2x^2 + 4x^3 + 7x^4 + 11x^5 + 16x^6 + 22x^7 + 29x^8 + 37x^9 + 46x^{10} + \cdots .$$

(1)



Proof. The structure is easily obtained by starting with the structure of 231-avoiders in Figure 1. From this structure we get the functional equation $E_2(x) = 1 + xE_2(x) + x^2D(x)^2$. Solving it gives the claimed equation for $E_2(x)$. □

We then consider three Wilf-equivalent bases.

Proposition 2 (Simion and Schmidt [1985], Propositions 7, 8 and 9)**.** *The structures of the permutation classes* $\mathcal{E}_{3,1}$*,* $\mathcal{E}_{3,2}$ *and* $\mathcal{E}_{3,3}$ *are given in Table 1. The enumeration is given by A011782 and the generating function is*

$$E_{3,1}(x) = \frac{1-x}{1-2x} = 1 + x + 2x^2 + 4x^3 + 8x^4 + 16x^5 + 32x^6 + 64x^7 + 128x^8 + 256x^9 + 512x^{10} + \cdots .$$

Proof. From the cover of $\mathcal{E}_{3,1}$ in Table 1 we obtain $E_{3,1}(x) = 1 + xD(x)E_{3,1}(x)$. Solving produces the claimed equation. For $\mathcal{E}_{3,2}$ we give two covers, the left-hand cover being similar to as above. For the right-hand cover, we have $\mathcal{I}$ and $\mathcal{D}$ mixing. This mixing is counted by the generating function $\frac{1}{1-2x}$ and we obtain $E_{3,2}(x) = 1 + \frac{x}{1-2x} = E_{3,1}(x)$. The cover of $\mathcal{E}_{3,3}$ is similar to the one of $\mathcal{E}_{3,1}$. □

Three patterns. The final basis we consider is counted by the Fibonacci numbers.

Proposition 3 (Simion and Schmidt [1985], Proposition 15)**.** *The structure of the permutation class* $\mathcal{F} = \mathrm{Av}(123, 132, 213)$ *is given by the cover in Equation* (2)*. They are enumerated by the Fibonacci numbers (A000045, shifted) and the generating function is*

$$F(x) = \frac{1}{1-x-x^2} = 1 + x + 2x^2 + 3x^3 + 5x^4 + 8x^5 + 13x^6 + 21x^7 + 34x^8 + 55x^9 + 89x^{10} + \cdots .$$

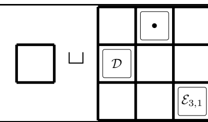
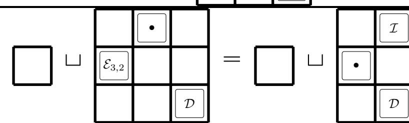
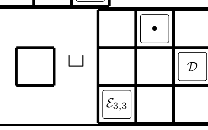
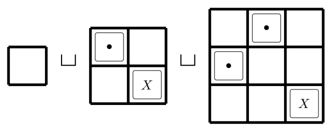
$\mathcal{E}_{3,1} = \text{Av}(123, 132)$	
$\mathcal{E}_{3,2} = \text{Av}(132, 312)$	
$\mathcal{E}_{3,3} = \text{Av}(231, 312)$	

TABLE 1. The structure of $\text{Av}(B)$ where B ranges over the non-symmetric bases in a Wilf-class containing two length three patterns

(2) 

Proof. From the cover, we obtain $F(x) = 1 + xF(x) + x^2F(x)$. Solving gives the claimed equation. □

4. ONE PATTERN OF LENGTH THREE AND ONE OF LENGTH FOUR

In this section, we consider bases consisting of one length three pattern and one length four pattern. These permutation classes were enumerated by West [1996] and Atkinson [1999]. We consider them here with an emphasis on an automated and uniform approach, and because we need to refer to some of them later.

There are eighteen non-symmetric cases, of which Struct can find a cover for sixteen[6]. The cover is easily verified, and a functional equation can be written for the generating function. As the methods are similar in all the proofs we only give the details in one case, for $\text{Av}(132, 4231)$ in Proposition 9.

Wilf-class 1. There is a single basis in this Wilf-class: $\{321, 1234\}$. There are finitely many avoiding permutations. Struct finds 21 rules, the largest of which is 6-by-6.

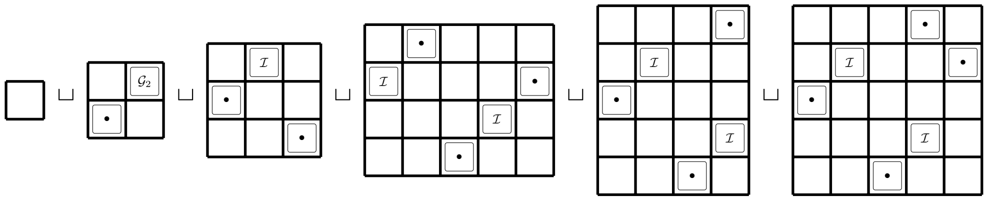
Wilf-class 2. There is a single basis in this Wilf-class.

Proposition 4 (Atkinson [1999], Proposition 3.1)**.** *The structure of the permutation class $\mathcal{G}_2 = \text{Av}(321, 2134)$ is given by the cover in Equation* (3)*. The enumeration is given by A116699 and the generating function is*

$$G_2(x) = \frac{1 - 4x + 7x^2 - 5x^3 + 3x^4 - x^5}{1 - 5x + 10x^2 - 10x^3 + 5x^4 - x^5} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 30x^5 + 61x^6 + 112x^7 + 190x^8 + 303x^9 + 460x^{10} + \cdots.$$

[6]The remaining two are treated in Section 7

(3)



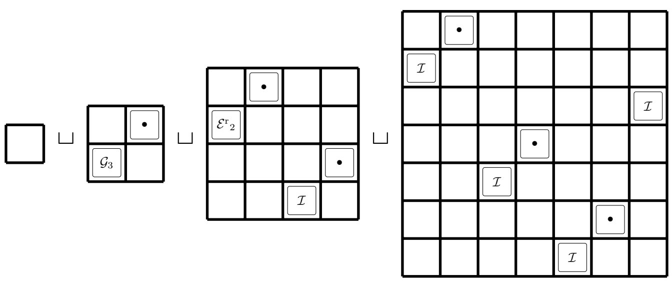
Proof. From the cover, we obtain $G_2(x) = 1 + xG_2(x) + x^2 I(x) + 2x^3 I(x)^3 + x^4 I(x)^4$. Solving gives the claimed equation. $\square$

Wilf-class 3. There is a single basis in this Wilf-class.

Proposition 5 (Atkinson [1999], Proposition 3.3)**.** *The structure of the permutation class* $\mathcal{G}_3 = \mathrm{Av}(132, 4321)$ *is given by the cover in Equation* (4)*. The enumeration is given by A116701 and the generating function is*

$$G_3(x) = \frac{1 - 4x + 7x^2 - 5x^3 + 3x^4}{1 - 5x + 10x^2 - 10x^3 + 5x^4 - x^5} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 31x^5 + 66x^6 + 127x^7 + 225x^8 + 373x^9 + 586x^{10} + \cdots .$$

(4)



Here $\mathcal{E}^{\mathrm{r}}{}_2 = \mathrm{Av}(132, 321)$ is the reverse of $\mathcal{E}_2$, and so they have the same enumeration.

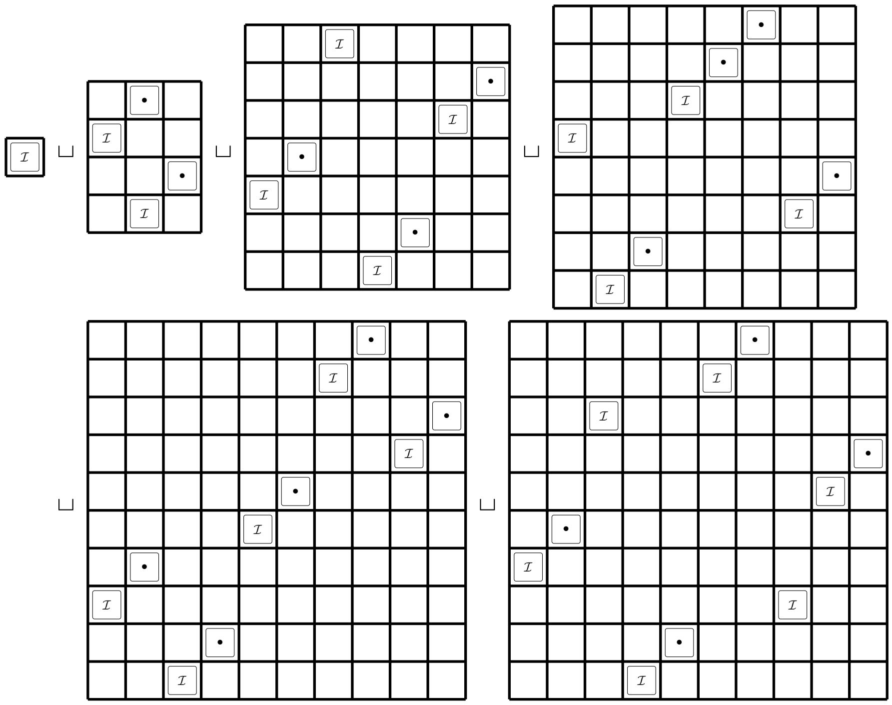
Proof. From the cover, we obtain $G_3(x) = 1 + xG_3(x) + x^2 E_2(x) I(x) + x^3 I(x)^4$. Solving produces the claimed equation. $\square$

Wilf-class 4. There is a single basis in this Wilf-class.

Proposition 6 (Atkinson [1999], Proposition 3.2)**.** *The structure of the permutation class* $\mathcal{G}_4 = \mathrm{Av}(321, 1324)$ *is given by the cover in Equation* (5)*. The enumeration is given by A179257 and the generating function is*

$$G_4(x) = \frac{1 - 5x + 11x^2 - 12x^3 + 8x^4 - 2x^5}{1 - 6x + 15x^2 - 20x^3 + 15x^4 - 6x^5 + x^6} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 32x^5 + 72x^6 + 148x^7 + 281x^8 + 499x^9 + 838x^{10} + \cdots .$$

(5)



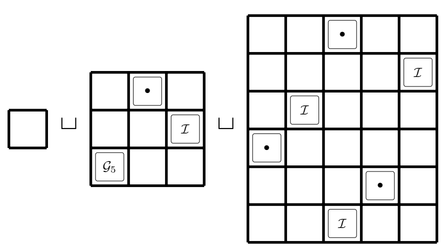
Proof. From the cover, we obtain $G_4(x) = I(x) + x^2 I(x)^3 + x^3 I(x)^4 + x^4 I(x)^4 + x^5 I(x)^5 + x^4 I(x)^6$. Solving produces the claimed equation. □

Wilf-class 5*.* There is a single basis in this Wilf-class.

Proposition 7 (West [1996])**.** *The structure of the permutation class* Av(321, 1342) *is given by the cover in Equation* (6)*. The enumeration is given by A116702 and the generating function is*

$$G_5(x) = \frac{1 - 4x + 6x^2 - 3x^3 + x^4}{1 - 5x + 9x^2 - 7x^3 + 2x^4} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 32x^5 + 74x^6 + 163x^7 + 347x^8 + 722x^9 + 1480x^{10} + \cdots .$$

(6)



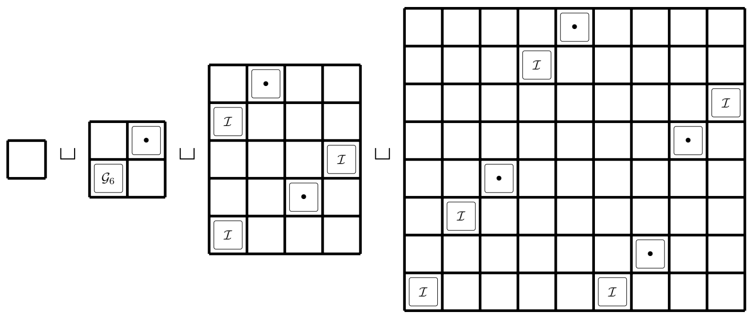
Proof. From the cover, we obtain $G_5(x) = 1 + xG_5(x)I(x) + x^3 I(x)^4$, and solving produces the claimed equation. □

Wilf-class 6. There is a single basis in this Wilf-class.

Proposition 8 (West [1996])**.** *The structure of the permutation class $\mathcal{G}_6 = \text{Av}(321, 2143)$ is given by the cover in Equation* (7)*. The enumeration is given by A088921 and the generating function is*

$$G_5(x) = \frac{1 - 4x + 5x^2 - x^3}{1 - 5x + 8x^2 - 4x^3} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 33x^5 + 80x^6 + 185x^7 + 411x^8 + 885x^9 + 1862x^{10} + \cdots .$$

(7)



Proof. From the cover, we obtain $G_6(x) = 1 + xG_6(x) + \frac{x^2 I(x)}{1-2x} + \frac{x^4 I(x)^3}{1-2x}$, and solving gives the claimed equation. □

Wilf-class 7. There are two bases in this Wilf-class enumerated by A005183 (shifted).

Proposition 9 (Atkinson [1999], Propositions 3.7 and 3.8 of supplement)**.** *The structures of the permutation classes $\mathcal{G}_{7,1} = \text{Av}(132, 4312)$ and $\mathcal{G}_{7,2} = \text{Av}(132, 4231)$ are given by the covers in Table 2. Their generating function is*

$$G_{7,1}(x) = \frac{1 - 4x + 5x^2 - x^3}{1 - 5x + 8x^2 - 4x^3} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 33x^5 + 81x^6 + 193x^7 + 449x^8 + 1025x^9 + 2305x^{10} + \cdots .$$

We will provide a proof for the cover obtained for $\mathcal{G}_{7,2}$. It illustrates the method needed for the other covers found in this section.

Proof. From the cover for $\mathcal{G}_{7,1}$ in Table 2 we obtain

$$G_{7,1}(x) = I(x) + \frac{x^2 I(x) E_{3,2}(x)}{1 - 2x}.$$

Solving produces the claimed equation. We will now provide proof for the cover of $\mathcal{G}_{7,2}$. A permutation of length $n \geqslant 1$ in $\mathcal{G}_{7,2}$ is of the form $\alpha n \beta$. If β is the empty permutation then α can be any permutation in $\mathcal{G}_{7,2}$. Otherwise, the permutation is of the form $\alpha n k \beta \gamma$ where all the letters in α are greater than those in β and γ, else you would have an occurrence of 132 using n, and all the elements in β are less than those in γ, else you would have an occurrence of 4231 using n and k. In order to avoid 132 and 4231, we have $\alpha \in \mathcal{E}_{3,2}$, $\beta \in \text{Av}(132, 231)$, and $\gamma \in \text{Av}(21)$. This is the cover for $\mathcal{G}_{7,2}$ in Table 2. From this cover we obtain

$$G_{7,2}(x) = 1 + xG_{7,2}(x) + x^2 I(x) E_{3,2}(x)^2,$$

and solving gives the claimed equation. □

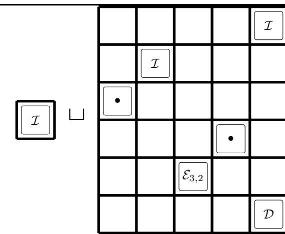
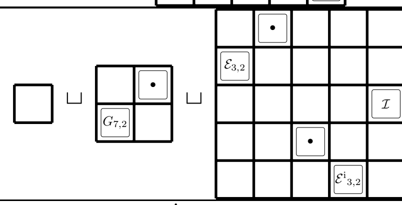
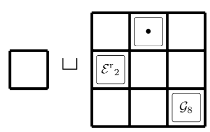
$\mathcal{G}_{7,1} = \text{Av}(132, 4312)$	
$\mathcal{G}_{7,2} = \text{Av}(132, 4231)$	

TABLE 2. The structure of $\mathcal{G}_{7,1}$ and $\mathcal{G}_{7,2}$. Here $\mathcal{E}^{\text{i}}{}_{3,2} = \text{Av}(132, 231)$ is the inverse symmetry of $\mathcal{E}_{3,2}$.

Wilf-class 8. There is a single basis in this Wilf-class.

Proposition 10 (Atkinson [1999], Proposition 3.9 of supplement)**.** *The structure of the permutation class* Av(132, 3214) *is given by the cover in Equation* (8)*. The enumeration is given by A116703 and the generating function is*

$$G_8(x) = \frac{1 - 3x + 3x^2 - x^3}{1 - 4x + 5x^2 - 3x^3} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 33x^5 + 82x^6 + 202x^7 + 497x^8 + 1224x^9 + 3017x^{10} + \cdots .$$

(8) 

Proof. From the cover, we obtain $G_8(x) = 1 + xE_2(x)G_8(x)$, and solving gives the claimed equation. □

Wilf-class 9. There are 9 bases in this Wilf-class, enumerated by a bisection of the Fibonacci numbers A001519.

Proposition 11 (Atkinson [1999], Propositions 16 and 18 of supplement)**.** *The structure of the permutation classes* $\mathcal{G}_{9,1} = \text{Av}(321, 3142)$*,* $\mathcal{G}_{9,2} = \text{Av}(132, 3412)$ *are given by the covers in Table 3. The generating function is*

$$G_{9,1}(x) = \frac{1 - 2x}{1 - 3x + x^2} = 1 + x + 2x^2 + 5x^3 + 13x^4 + 34x^5 + 89x^6 + 233x^7 + 610x^8 + 1597x^9 + 4181x^{10} + \cdots .$$

Proof. From the covers, we obtain

$$G_{9,1}(x) = 1 + xG_{9,1}(x) + \frac{x^2 G_{9,1}}{1 - 2x}.$$

Solving gives the claimed equation. □

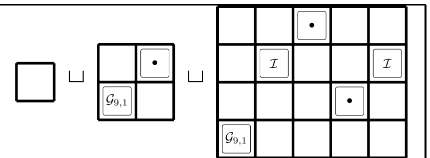
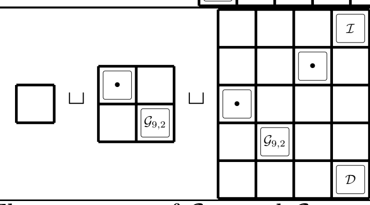
Av(321, 3142)	
Av(132, 3412)	

TABLE 3. The structure of $\mathcal{G}_{9,1}$ and $\mathcal{G}_{9,2}$

Proposition 12 (Atkinson [1999], Propostions 11, 12, 13, 14 and 17 of supplement)**.** *The structures of the permutation classes* $\mathcal{G}_{9,3} = \mathrm{Av}(132, 1234)$*,* $\mathcal{G}_{9,4} = \mathrm{Av}(132, 4213)$*,* $\mathcal{G}_{9,5} = \mathrm{Av}(132, 4123)$*,* $\mathcal{G}_{9,6} = \mathrm{Av}(132, 3124)$ *and* $\mathcal{G}_{9,7} = \mathrm{Av}(132, 2134)$ *are given by the covers in Table 4. The generating function is equal to* $G_{9,1}(x)$*.*

Proof. From the covers, we obtain $G_{9,3}(x) = 1 + xE_{3,1}(x)G_{9,3}(x)$, and solving gives the claimed equation. □

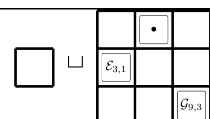
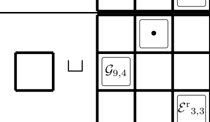
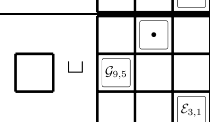
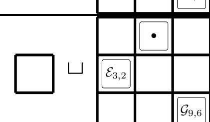
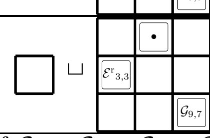
Av(132, 1234)	
Av(132, 4213)	
Av(132, 4123)	
Av(132, 3124)	
Av(132, 2134)	

TABLE 4. The structure of $\mathcal{G}_{9,3}$, $\mathcal{G}_{9,4}$, $\mathcal{G}_{9,5}$, $\mathcal{G}_{9,6}$ and $\mathcal{G}_{9,7}$

There are two more bases in this Wilf-class, $\mathcal{G}_{9,8} = \text{Av}(321, 2341)$ and $\mathcal{G}_{9,9} = \text{Av}(321, 3412)$. We have not been able to find a cover for these permutation classes. The reason for this appears linked to the reason there is no cover for Av(123). We will revisit these permutation classes in Section 7, in particular, Proposition 13.

5. Comparison with existing approaches

5.1. **Polynomial permutation classes.** A permutation class $\mathcal{C}$ is said to be *polynomial* if the number of length n permutations, $|\mathcal{C}_n|$, is given by a polynomial for all sufficiently large n. One of the first general results on permutation classes by Kaiser and Klazar [2002/03] states that if the number of length n permutations in a permutation class is less than the nth Fibonacci number then the permutation class is polynomial. This is known as the Fibonacci dichotomy and alternative proofs were given by Huczynska and Vatter [2006] and Albert et al. [2007]. From the results of Homberger and Vatter [2016], we get the following theorem.

Theorem 1. *All polynomial permutation classes have a cover.*

In order to prove this, we will recall some definitions used by Homberger and Vatter [2016]. A *peg permutation* is a permutation where each letter is decorated with a $+$, $-$ or $\circ$, for example, $\rho = 3^\circ 1^- 4^\circ 2^+$. Let M_ρ be the matrix defined by

$$M_{i,j} = \begin{cases} \text{Av}(12) & \text{if } \rho_i = j^+ \\ \text{Av}(21) & \text{if } \rho_i = j^- \\ \{1\} & \text{if } \rho_i = j^\circ \end{cases}$$

then $\text{Grid}(\rho) = \text{Grid}(M_\rho)$.

A peg permutation is, therefore, a geometric grid class with monotone intervals for its matrix entries. We can specify these intervals with vectors on $\mathbb{N}$, the non-negative integers. We call this a *ρ-partition*. For example, we could write

$$6321745 = 3^\circ 1^- 4^\circ 2^+[\langle 1, 3, 1, 2\rangle].$$

Throughout it is insisted that we use vectors that *fill* peg permutations, meaning that a component of a vector equals 1 if it corresponds to a $\circ$ and otherwise is at least 2. For a set of filling vectors $\mathcal{V}$, define

$$\rho[\mathcal{V}] = \{\rho[v] : v \in \mathcal{V}\}.$$

Given vectors v and w in $\mathbb{N}^m$, then v is contained in w if $v(i) \leqslant w(i)$ for all indices i. This is a partial order and moreover, if v is contained in w then for a length m peg permutation, $\rho[v]$ is contained in $\rho[w]$, assuming this is defined. A set closed downwards under containment is called a *downset*, and closed upwards an *upset*. The intersection of a downset and an upset is called a *convex set*.

The set of vectors which fill a given peg permutation ρ forms a convex set. The downset component of this convex set consists of those vectors which do not contain an entry larger than 1 corresponding to a dotted entry of ρ. The upset component consists of those vectors which contain the vector v defined by $v(i) = 1$ if $\rho(i)$ is dotted and $\rho(i) = 2$ if $\rho(i)$ is signed. As we discussed in Section 1, all polynomial permutation classes can be represented by a finite set of peg permutations. In fact, a more general condition holds.

We can now state the result from [Homberger and Vatter, 2016, Theorem 1.4].

Theorem 2 (Homberger and Vatter [2016], Theorem 1.4 and Proposition 2.3)**.** *For every polynomial permutation class $\mathcal{C}$ there is a finite set H of peg permutations, each associated with its own convex set $\mathcal{V}_\rho$ of filling vectors, such that $\mathcal{C}$ can be written as the disjoint union*

$$\mathcal{C} = \bigsqcup_{\rho \in H} \rho[\mathcal{V}_\rho].$$

In [Homberger and Vatter, 2016, Proposition 2.3] the authors show that every permutation which fills a peg permutation ρ has a unique ρ-partition. Together with Theorem 2, this leads to the following result.

Theorem 3. *A peg permutation ρ and its convex set $\mathcal{V}_\rho$ of filling vectors is a* Struct *rule.*

Proof. For a peg permutation, ρ, V_ρ consists of the vectors with $\rho(i) = 1$ when $v(i)$ is dotted and the remaining elements an integer greater than 1. We create ρ' from ρ by replacing an entry x^+ in ρ with the subsequence $(x-0.2)^\circ(x-0.1)^\circ x^+$ and entries y^- with the subsequence $(y+0.2)^\circ(y+0.1)^\circ y^-$ and taking the standardization of the underlying permutation. The set $\mathrm{Grid}(M_{\rho'})$ is a generalized grid class. Moreover, it is a Struct rule since every permutation which fills ρ has a unique ρ-partition. □

Theorem 1 follows as a corollary, as we now have a finite set of Struct rules. Given there exists a cover for all polynomial permutation classes a natural follow-up question is: for a polynomial permutation class $\mathcal{C}$ is there a bound on the size of the Struct rules required in a cover for $\mathcal{C}$?

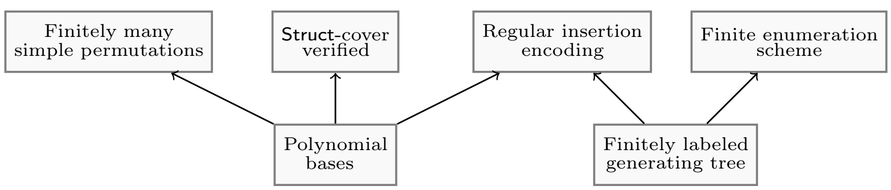


Figure 2. Comparison of enumeration methods

5.2. **Comparison with other algorithms.** In the remainder of this section, we collect examples of permutation classes which separate the automatic methods mentioned in the introduction. Regular insertion encodable permutation classes are always rational. This implies that Av(231) does not have a regular insertion encoding. As we saw in Section 1 this permutation class has a Struct cover. It has a finite enumeration scheme detailed by [Vatter, 2008, p. 7-10] and has finitely many simple permutations and, therefore, [Albert and Atkinson, 2005, Theorem 8] tells us it can be enumerated using the substitution decomposition. For Av(123) only finite enumeration schemes have success, given in plain English by [Zeilberger, 1998, p. 3-5].

The permutation class Av(123, 3214) has a finitely labeled generating tree given by [Vatter, 2006, p. 10], but we have not found a Struct cover for it. The permutation class Av(321, 2143), however, does have a cover shown in Section 4, although it can not be enumerated using any of the other automatic methods.

We have from Theorem 1 that all polynomial permutation classes have a cover. They are also regular insertion encodable. With a quick check (for example using the conditions given by [Albert et al., 2007, Theorem 1]) we see that Av(1234, 4231) is a polynomial permutation class, therefore is regular insertion encodable and has a cover. As was shown by [Vatter, 2008, Proposition 7.1] this permutation class does not have an enumeration scheme.

The separable permutations Av(2413, 3142) have finitely many simple permutations and so by [Albert and Atkinson, 2005, Theorem 8] they can be enumerated using the substitution decomposition. This permutation class can not be enumerated using any of the other methods.

basis	Polynom.	Fin. gen. tree	Reg. ins. enc.	Struct cover	Fin. enum. scheme	Fin. many simple perms.
132				✓	✓	✓
123					✓	
123, 3214		✓	✓		✓	
321, 2143				✓		
1234, 4231	✓		✓	✓		✓
2413, 3142						✓

Table 5. Bases separating the methods

6. Patterns of length four

The enumeration and Wilf-classification are known for all permutation classes with a subset of $\mathcal{S}_3$ for a basis. As mentioned, there exists a Struct cover for all of these classes except Av(123), which we revisit in Section 7.

We will now look at the results of applying the algorithm to bases containing length four patterns. The total number of bases is 2^{24}, but of course, it suffices to look at one basis from each symmetry class. This brings the total down to $2\,097\,152 \approx 2^{21}$ bases.[7] In Section 5 we showed that all polynomial permutation classes have a cover. We, therefore, look at the non-polynomial classes, bringing the total down to $157\,736 \approx 2^{17}$ bases.

As is to be expected Struct does better on bases with many patterns. In Table 6 we break down the computer runs by the size of the basis and the size of the largest dimension of the Struct rules required for the cover.[8] All of the non-polynomial bases were tried with a cover with maximum rule size of 7×7. We consider the permutation class failed if Struct does not find a cover within this bound. Of course, there may exist covers using larger rules.

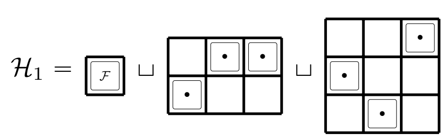
Judging from the data in the table, one would have hoped to get some successes for bases with three patterns. However, at this point the memory usage becomes infeasible for the computers we have access to, routinely exceeding 32GiB of memory. We, therefore, expect there to be some such permutation classes with covers consisting of 7×7 rules, although we can not find them at this point.

6.1. **A collection of successes.** The longest basis of length four patterns corresponding to a non-polynomial permutation class has 19 patterns

$$\mathcal{H}_1 = \text{Av}(1234, 1243, 1324, 1342, 1423, 1432, 2134, 2143, 2314, 2341, 2413, 2431, 3124, 3142, 3214, 3241, 4123, 4132, 4213).$$

Struct finds the cover

(9)



[7]In general one can count subsets of S_n with respect to the symmetries of the square. This was added to the OEIS by the authors (A277086). It follows from a simple application of Burnside's Lemma.

[8]Most of the computation was done on a cluster owned by Reykjavik University, and the remainder on a cluster owned by the University of Iceland.

				successes					
$\|B\|$	non-symmetric	non-polynomial	failures	2×2	3×3	4×4	5×5	6×6	7×7
24	1	0	0	0	0	0	0	0	0
23	7	0	0	0	0	0	0	0	0
22	56	0	0	0	0	0	0	0	0
21	317	0	0	0	0	0	0	0	0
20	1524	0	0	0	0	0	0	0	0
19	5733	1	0	0	1	0	0	0	0
18	17728	9	0	0	8	1	0	0	0
17	44767	58	0	0	32	26	0	0	0
16	94427	285	0	0	75	206	4	0	0
15	166786	1069	0	0	118	901	49	1	0
14	249624	3143	0	0	137	2620	377	9	0
13	316950	7338	0	0	122	5118	2038	60	0
12	343424	13891	0	1	82	6372	7163	273	0
11	316950	21451	1	0	36	4890	15551	970	3
10	249624	27274	12	0	9	2285	21947	2990	31
9	166786	28391	59	0	1	615	19672	7856	188
8	94427	24160	177	6	0	85	9956	13051	885
7	44767	16489	708	0	0	10	2267	10924	2580
6	17728	8935	3249	0	0	2	167	3668	1849
5	5733	3716	2597	0	0	0	7	331	781
4	1524	1187	1160	3	0	0	1	8	15
3	317	279	279	0	0	0	0	0	0
2	56	53	53	0	0	0	0	0	0
1	7	7	7	0	0	0	0	0	0

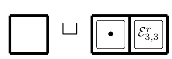
TABLE 6. Data from running Struct on every non-polynomial basis with length four patterns

where $\mathcal{F} = \text{Av}(123, 132, 213)$. We, therefore, see that $\mathcal{H}_1$ consists of permutations avoiding $123, 132, 213$, with the addition of those same permutations. This can also be observed directly from the basis of $\mathcal{H}_1$: it consists of all length four patterns containing at least one of the patterns in the basis of $\mathcal{F}$.

In the column corresponding to size 2×2 rules, we see three successful bases with 4 patterns, six bases with 8 patterns and one basis with 12 patterns. All of the covers found are similar, e.g., one of the size 8 bases is

$$\mathcal{H}_2 = \text{Av}(1243, 1324, 2143, 2314, 3142, 3214, 4132, 4213)$$

and has the cover

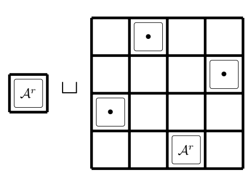
(10) $\mathcal{H}_2 =$ 

where $\mathcal{E}^r_{3,3} = \text{Av}(132, 213)$. The enumeration is therefore given by $n2^{n-2}$. The covers for the three successful bases with 4 patterns are essentially the same as some of the structures discussed in Bruner [2015].

There are thirteen conjectures given by Kuszmaul [2015] about the enumeration of bases containing many length four patterns, of which ten have a regular insertion encoding. Struct found covers for these as well as one of the remaining three:

$$\mathcal{H}_3 = \text{Av}(2431, 2143, 3142, 4132, 1432, 1342, 1324, 1423, 1243)$$

Struct found the cover

(11) $\mathcal{H}_3 =$ 

where $\mathcal{A}^r = \mathrm{Av}(132)$. This leads to the equation $H_3(x) = A(x) + x^3A(x)$ (A071742), where $A(x)$ is the generating function for the Catalan numbers.

Although it is not a focus of this paper we show how this cover can be used to enumerate the class with respect to some permutation statistics. If we let y track the number of left-to-right-minima,

$$H_3(x, y) = A(x, y) + x^3yA(x, y).$$

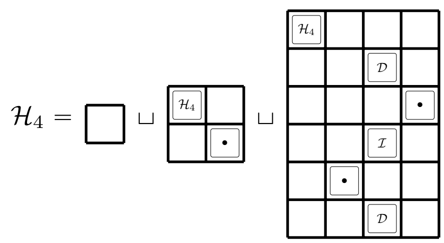
Then if we let z track the number of inversions then

$$H_3(x, z) = A(x, z) + x^3zA(xz^2, z).$$

Finally, if we let w track the number of descents then

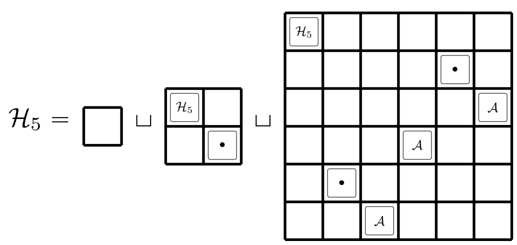
$$H_3(x, w) = A(x, w) + x^3A(x, w).$$

To highlight some interesting structures found by Struct we end this section by showing the covers found for three permutation classes with a basis consisting of four length four patterns. For the permutation class $\mathcal{H}_4 = \mathrm{Av}(1324, 1342, 3124, 3142)$ the cover found was



which leads to the equation $H_4(x) = 1 + xH_4(x) + \frac{x^2H_4(x)}{1-3x}$ and upon solving gives $H_4(x) = \frac{1-3x}{1-4x+2x^2}$ (A006012, shifted).

The permutation class $\mathcal{H}_5 = \mathrm{Av}(1342, 2314, 2413, 3142)$ was enumerated by [Atkinson and Stitt, 2002, Theorems 18 and 19] using methods with the wreath product. Struct finds the cover



where $\mathcal{A} = \mathrm{Av}(231)$. This leads to the equation $H_5(x) = 1 + xH_5(x) + x^2H_5(x)A(x)^3$ and to the same function first found by Atkinson and Stitt [2002] (A078483).

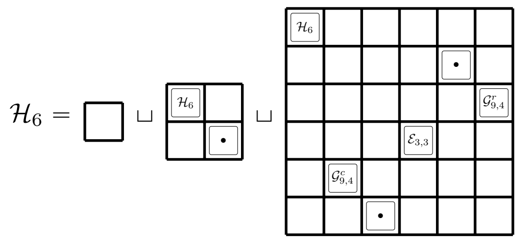
Choosing a permutation at random from a permutation class is, in general, a very difficult problem. For example, the problem of what does a typical permutation in a class look like was initiated by Madras and Liu [2010], and, for example, further studied by Miner and Pak [2014] and Bassino et al. [2015].

We do not address this problem in any generality here but for the cover for $\mathcal{H}_5$ above we can use the following method: first, solve the functional equation for $H_5(x)$ above to obtain an enumeration of all the permutations in the class. Then find the enumeration of the permutations in the grid classes of the individual rules. These are, ignoring the empty rule, $[x^n]xH_5(x)$ and $[x^n]x^2A(x)^3H_5(x)$. These give us weights for a probability distribution between the rules. Then we need to be able to choose a permutation uniformly at random from a given rule. We only look at the larger rule here. We use variables a, b, c, d, to track the number of points in each of the blocks containing a permutation class, read from left to right. This gives the generating function $x^2 H_5(ax)A(bx)A(cx)A(dx)$. The probability of choosing a permutation with i, j, k, ℓ in each of the blocks is now seen to be

$$\frac{[x^n a^i b^j c^k d^\ell] x^2 H_5(ax)A(bx)A(cx)A(dx)}{[x^n] x^2 A(x)^3 H_5(x)}.$$

We then need to recursively apply this method to choose a length i permutation from $\mathcal{H}_5$ and choose random permutations of lengths j, k, ℓ from $\mathcal{A}$. This method was formalized by Flajolet et al. [1994].

The enumeration of the permutation class $\mathcal{H}_6 = \mathrm{Av}(1342, 2413, 3124, 3142)$ is not on the OEIS. The cover found by Struct is



where $\mathcal{G}^c_{9,4} = \mathrm{Av}(312, 1342)$ and $\mathcal{G}^r_{9,4} = \mathrm{Av}(231, 3124)$. This can be easily verified and leads to the equation $H_6(x) = 1 + xH_6(x) + xE_{3,1}(x)G_{9,1}(x)^2$. Solving gives $H_6(x) = \frac{1-6x+11x^2-6x^3+x^4}{1-7x+16x^2-14x^3+5x^4-x^5}$.

This cover is typical of many found by Struct, in that it places into the rule a permutation class of the form $\mathrm{Av}(P, Q)$ where P is a non-empty subset of $\mathcal{S}_3$ and Q is a subset of $\mathcal{S}_4$.[9] It is known that all such classes have a rational generating function, see Albert et al. [2016] and Albert and Atkinson [2005], but they have not been computed. We believe that these should all be computed as a first step in attempting to enumerate all permutation classes containing length four patterns.

6.2. **Discussion of the failures.** The permutation class with the largest basis that Struct fails to find a cover for using 7×7 rules, has 11 patterns, namely

$$\mathrm{Av}(1234, 1243, 1324, 1342, 1423, 1432, 2134, 2143, 2314, 2341, 3214).$$

It can, however, be done with the same methods as are needed for the Struct failures we discuss in Section 7. This leads to the generating function $\frac{1-x+x^3}{1-2x-x^3}$.

After the first failure, the failure rate stays below 1% for bases with at least 8 patterns, rising to 5% for 7 patterns, 37% for 6 patterns, 70% for 5 patterns, 98% for 4 patterns, until every basis with 3 or fewer patterns fails.

7. Further improvements

Given a conjectured cover from Struct for a permutation class $\mathcal{C}$, it is natural to want to check that this is, in fact, a cover for $\mathcal{C}$. Given an individual Struct rule it can be verified automatically if it generates a subset of a permutation class $\mathcal{C}$: If there is a permutation π in the Struct rule $\mathcal{R}$ which contains a basis element of length k, remove all the non-points from the gridding of the permutation so that the permutation still contains an occurrence of the basis element. This reduced permutation has at most $\ell + k$ points. Therefore if $\mathcal{R}_{\leqslant \ell+k} \subseteq \mathcal{C}$, where k is the length of the longest basis element, then $\mathcal{R} \subseteq \mathcal{C}$. This brings us closer to an automatic check if a cover found by Struct

[9] Up to symmetries there are 14 181 such bases.

is guaranteed to be a subset of the permutation class $\mathcal{C}$. What remains is there might still be an overlap between the permutations generated by two rules. Below we sketch how the steps one usual performs to verify the cover by hand, can be turned into an algorithm.

7.1. **Proof Trees.** To motivate the section we revisit the proof of Proposition 9, concerning the permutation class $\mathcal{G}_{7,2} = \mathrm{Av}(132, 4231)$, and turn the verification of the cover into a "proof tree" in Figure 3.

We start with a root vertex representing $\mathcal{G}_{7,2}$. The empty permutation is in $\mathcal{G}_{7,2}$, and we add a left child of the root indicating this. All other permutations have a topmost point and we add the right child to represent them. This is analogous as to saying a non-empty permutation can be written as $\alpha n \beta$. In our tree, we observe that the right part must avoid 231, and so be in $\mathrm{Av}(132, 231)$. We continue with our argument, for permutations where β is empty, α can be any permutation in $\mathcal{G}_{7,2}$ so we indicate this by adding a left child of the node we are on. Otherwise, β has a leftmost point, and we add a right child to represent them. We observe that the points to the right and above the leftmost point must be in $\mathrm{Av}(21)$, and to the right and below must be in $\mathrm{Av}(132, 231)$. We have now reached the left tree in Figure 3.

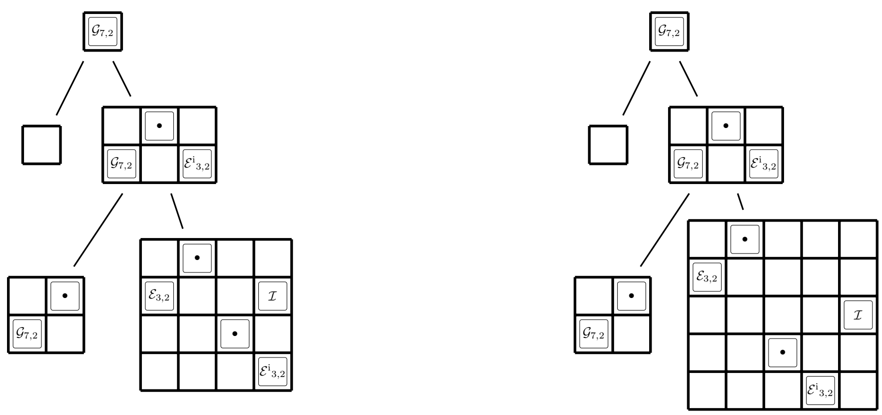


FIGURE 3. The structure of $\mathrm{Av}(132, 4231)$.

The only "proof strategy" (PS) used so far is

(PSa) Let v be a leaf with some cell c containing a non-empty permutation class. Create two children v^ℓ, v^r of v where in v^ℓ the empty permutation was chosen in c; and in v^r an extreme point (right, top, left or bottom) has been inserted. Infer as much information as possible about the newly created generalized grid class rule in v^r.

We continue with our example, and observe that there can not be a non-inversion between the cells containing $\mathcal{G}_{7,2}$ and $\mathcal{I}$, because an occurrence of 132 would be formed. We can, therefore, draw a horizontal line between these cells and separate them. We also observe there can not be an inversion between the cells containing $\mathcal{I}$ and $\mathcal{E}^{\mathrm{i}}{}_{3,2}$, otherwise, an occurrence of 4231 would be formed. We can, therefore, draw a vertical line between these cells and separate them. We have now reached the right tree in Figure 3.

The proof strategy we used was

(PSb$^{\text{ri}}$) Let v be a leaf with only two non-empty cells s and t in a particular row. If an inversion crossing between s and t creates a pattern in the basis, draw a horizontal line splitting the row. Let $s_\uparrow$, $s_\downarrow$, and $t_\uparrow$, $t_\downarrow$, be the cells s and t split into, then $s_\uparrow = t_\downarrow = \epsilon$, and infer as much information as possible about the newly created cells.

There are three other versions of this proof strategy, about cells in the same column and insertion of a non-inversion. We collectively call them (PSb).

Applying the proof strategies (PSa) and (PSb) creates trees with the property that the disjoint union of children is a superset of their parent. We will refer to this as the *superset property*.

From the superset property, it follows that at each stage the leaves form a superset, in the form of a disjoint union, of the root. Leaves that are subsets of the root, i.e., avoid the patterns in the basis will be called *verified*. If every leaf is verified, then the root is a disjoint union of the sets represented by the leaves.

For all the covers seen in this paper so far, it is possible to draw a similar proof tree using (PSa) and (PSb). We will now look at bases Struct did not find a cover for. In order to find proof trees, we need a new proof strategy.

(PSc) Let $\text{Av}(B)$ be the set under consideration. Let v be a non-verified leaf in a tree. Let v' be a subset of cells in v satisfying the following two conditions:
  (1) When every other cell is deleted, what remains is an ancestor vertex u of v.
  (2) Let c be a placement of points in u resulting in a permutation in $\text{Av}(B)$ and let c' be the corresponding placement of points in v'. Then any completion of the placement c' to a placement on v results in a permutation in $\text{Av}(B)$.

  We declare the leaf as verified and modify the permutations represented by v by using only placements of the form defined in condition (2).

When (PSc) is applied to a subset v' of a vertex v we add a dotted edge from the ancestor vertex u to v. This is not a part of the tree, just to keep track of where (PSc) has been applied. With our new proof strategy, we now revisit some bases that Struct did not find a cover for.

7.2. **One pattern of length three and one of length four - revisited.** In Section 6 we found covers for almost all of the bases with one length three pattern and one length four pattern. The covers we found can all be converted into proof trees. There were two bases that Struct could not find a cover for. There does exist a proof tree for these bases. Both arguments are similar and we will only provide a proof for the structure found in the first case.

Proposition 13 (West [1996]). *The structures of the permutation classes* $\mathcal{G}_{9,8} = \text{Av}(321, 2341)$ *and* $\mathcal{G}_{9,9} = \text{Av}(321, 3412)$ *are given by the proof trees in Figure 4. Their generating functions are equal to* $G_{9,1}(x)$.

Proof. The root node is $\mathcal{G}_{9,8}$. We first use (PSa) on the root node, choosing the topmost point. The left child is verified. On the right child, we use (PSa) on the cell containing $\mathcal{I}$, choosing the leftmost point. The left child is verified. On the right child, we use (PSa) on the node containing $\mathcal{I}$ in the far left column, choosing the rightmost. The left child is verified. On the right child, we use (PSb) on the two cells in the rightmost column containing $\mathcal{I}$. We then use (PSc) from the right child of the root vertex to the current node. All leaves are now verified. We now have the left proof tree in Figure 4. From this proof tree, we obtain

$$G_{9,8}(x) = 1 + xG_{9,8}(x) + x^2D(x)G_{9,8}(x) + x^2D(x)^2(G_{9,8}(x) - 1).$$

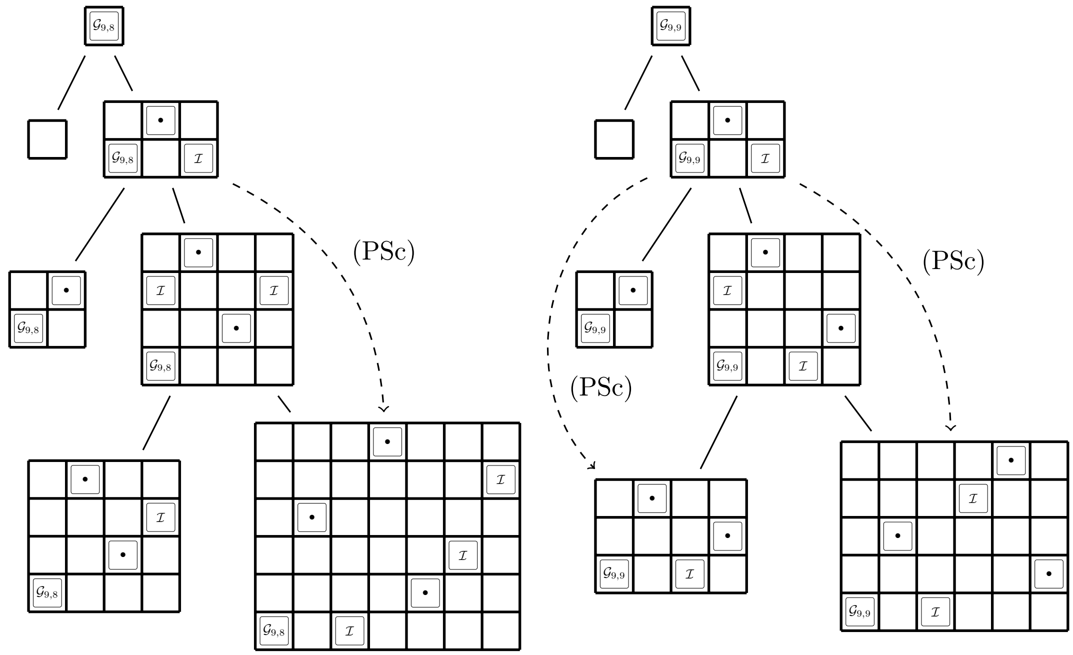


FIGURE 4. The structures of $\mathcal{G}_{9,8} = \mathrm{Av}(321, 2341)$ and $\mathcal{G}_{9,9} = \mathrm{Av}(321, 3412)$.

Solving gives the claimed equation. From the proof tree on the right in Figure 4, we obtain

$$G_{9,9}(x) = 1 + xG_{9,9}(x) + x(G_{9,9}(x) - 1) + x^2 D(x)(G_{9,9}(x) - 1).$$

Solving gives the claimed equation. □

Therefore with the addition of (PSc), all bases with one length three and one length four pattern have a proof tree.

7.3. **Patterns of length three - revisited.** In Section 3 we found covers for all bases with patterns of length three, except {123}. There does exist a proof tree for Av(123).

Proposition 14. *The structure of the permutation class* $\mathcal{A}_2 = \mathrm{Av}(123)$ *is given by the proof tree in Figure 5.*

We omit the proof as it follows similar arguments as in the previous proposition. From the proof tree in Figure 5, it is easy to derive a recurrence for enumerating Av(123). It is not trivial to show that this enumeration is equal to the Catalan numbers. We are therefore still lacking more powerful algorithms for a fully automatic Wilf-classification of all bases in $\mathcal{S}_3$.

8. CONCLUSION

It is decidable whether a basis defines a polynomial permutation class. The same is true for regular insertion encodable permutation classes. Given that polynomial permutation classes have covers, it is natural to ask the following.

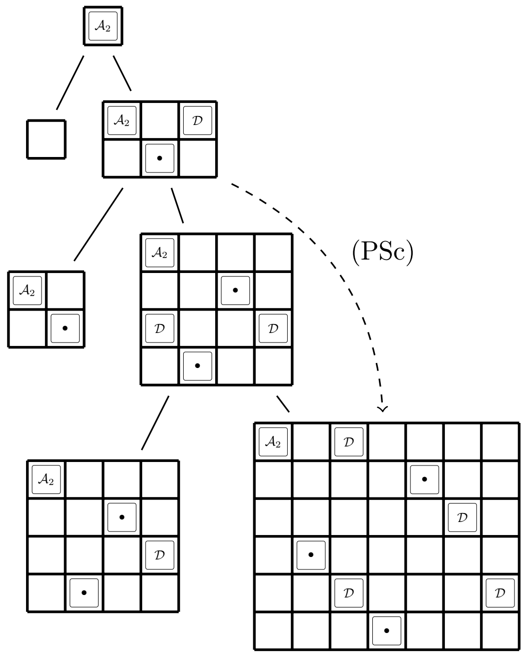


FIGURE 5. The structure of $\mathcal{A}_2 = \mathrm{Av}(123)$.

Question 1. Given a basis B, is it decidable whether a cover exists for the permutation class $\mathrm{Av}(B)$? Moreover, is there a bound on the size of the rules required?

There is no known bound on the size of the peg permutations (and therefore Struct rules) for polynomial permutation classes, suggesting that this question might be hard to answer. In Section 7 we showed that it is easy to verify if a Struct rule is a subset of a permutation class $\mathcal{C}$.

Question 2. Can you verify when two rules are disjoint?

Given this, we would then be able to show that a Struct cover is a subset of the permutation class. Of course, the goal is to answer the following.

Question 3. Given a finite set of Struct rules, is it decidable if these form a cover for a permutation class?

This is a natural question to ask, and the proof trees discussed in Section 7 go some way to answering this, however, it is not clear that all covers can be made into a proof tree argument.

REFERENCES

M. H. Albert and M. D. Atkinson. Simple permutations and pattern restricted permutations. *Discrete Math.*, 300(1-3):1–15, 2005.

M. H. Albert, M. D. Atkinson, and Robert Brignall. Permutation classes of polynomial growth. *Ann. Comb.*, 11(3-4):249–264, 2007.

M. H. Albert, Robert Brignall, Nik Ruškuc, and Vincent Vatter. Rationality for subclasses of 321-avoiding permutations. *arXiv:1602.00672*, 2016.

Michael H. Albert, Steve Linton, and Nik Ruškuc. The insertion encoding of permutations. *Electron. J. Combin.*, 12:Research Paper 47, 31, 2005.

Michael H. Albert, M. D. Atkinson, Mathilde Bouvel, Nik Ruškuc, and Vincent Vatter. Geometric grid classes of permutations. *Trans. Amer. Math. Soc.*, 365(11):5859–5881, 2013.

Arnar Arnarson, Christian Bean, Alfur Bjarnason, Unnar Erlendsson, Henning Ulfarsson, and Sigurjon Viktorsson. PermPAL. `http://permpal.ru.is/`, 2017.

M. D. Atkinson. Restricted permutations. *Discrete Math.*, 195(1-3):27–38, 1999.

M. D. Atkinson and T. Stitt. Restricted permutations and the wreath product. *Discrete Math.*, 259(1-3):19–36, 2002.

Frédérique Bassino, Mathilde Bouvel, Adeline Pierrot, and Dominique Rossin. Deciding the finiteness of the number of simple permutations contained in a wreath-closed class is polynomial. *Pure Math. Appl. (PU.M.A.)*, 21(2):119–135, 2010.

Frédérique Bassino, Mathilde Bouvel, Adeline Pierrot, Carine Pivoteau, and Dominique Rossin. An algorithm computing combinatorial specifications of permutation classes. *arXiv:1506.00868*, 2015.

Christian Bean, Bjarki Gudmundsson, and Henning Ulfarsson. PermStruct. `https://github.com/PermutaTriangle/PermStruct`, 2017.

Armin Biere. Lingeling, Plingeling and Treengeling entering the SAT competition, 2013.

Marie-Louise Bruner. Central binomial coefficients also count (2431, 4231, 1432, 4132)-avoiders. 2015.

Philippe Flajolet, Paul Zimmerman, and Bernard Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoret. Comput. Sci.*, 132(1-2):1–35, 1994.

Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2016. URL `http://www.gurobi.com`.

Cheyne Homberger and Vincent Vatter. On the effective and automatic enumeration of polynomial permutation classes. *J. Symbolic Comput.*, 76:84–96, 2016.

Sophie Huczynska and Vincent Vatter. Grid classes and the Fibonacci dichotomy for restricted permutations. *Electron. J. Combin.*, 13(1):Research Paper 54, 14 pp. (electronic), 2006.

Tomáš Kaiser and Martin Klazar. On growth rates of closed permutation classes. *Electron. J. Combin.*, 9(2):Research paper 10, 20, 2002/03. Permutation patterns (Otago, 2003).

D.E. Knuth. *The art of computer programming. Vol. 3.* Addison-Wesley, Reading, MA, 1998. Sorting and searching, Second edition.

William Kuszmaul. Fast algorithms for finding pattern avoiders and counting pattern occurrences in permutations. *CoRR*, abs/1509.08216, 2015.

Neal Madras and Hailong Liu. Random pattern-avoiding permutations. In *Algorithmic probability and combinatorics*, volume 520 of *Contemp. Math.*, pages 173–194. Amer. Math. Soc., Providence, RI, 2010.

Sam Miner and Igor Pak. The shape of random pattern-avoiding permutations. *Adv. in Appl. Math.*, 55:86–130, 2014.

Rodica Simion and Frank W. Schmidt. Restricted permutations. *European J. Combin.*, 6(4): 383–406, 1985.

N. J. A. Sloane. The on-line encyclopedia of integer sequences, published electronically at http://www.oeis.org, 2017.

Vincent Vatter. Finitely labeled generating trees and restricted permutations. *J. Symbolic Comput.*, 41(5):559–572, 2006.

Vincent Vatter. Enumeration schemes for restricted permutations. *Combin. Probab. Comput.*, 17 (1):137–159, 2008.

Vincent Vatter. Small permutation classes. *Proc. Lond. Math. Soc. (3)*, 103(5):879–921, 2011.

Vincent Vatter. Finding regular insertion encodings for permutation classes. *J. Symbolic Comput.*, 47(3):259–265, 2012.

Julian West. Generating trees and forbidden subsequences. In *Proceedings of the 6th Conference on Formal Power Series and Algebraic Combinatorics (New Brunswick, NJ, 1994)*, volume 157, pages 363–374, 1996.

Doron Zeilberger. Enumeration schemes and, more importantly, their automatic generation. *Ann. Comb.*, 2(2):185–195, 1998.