



Contents lists available at ScienceDirect

Journal of Symbolic Computation

journal homepage: www.elsevier.com/locate/jsc

An algorithmic approach based on generating trees for enumerating pattern-avoiding inversion sequences



Ilias Kotsireas^a, Toufik Mansour^b, Gökhan Yıldırım^{c,1}

^a CARGO Lab, Wilfrid Laurier University, 75 University Avenue West, Waterloo, Ontario N2L 3C5, Canada

^b Department of Mathematics, University of Haifa, 3498838 Haifa, Israel

^c Department of Mathematics, Bilkent University, 06800 Ankara, Turkey

ARTICLE INFO

Article history:

Received 8 June 2022

Received in revised form 12 May 2023

Accepted 15 May 2023

Available online 18 May 2023

MSC:

05A05

05A15

05A16

Keywords:

Pattern-avoiding inversion sequences

Generating functions

Generating trees

Kernel method

Catalan numbers

Motzkin numbers

ABSTRACT

We introduce an algorithmic approach based on a generating tree method for enumerating the inversion sequences with various pattern-avoidance restrictions. For a given set of patterns, we propose an algorithm that outputs either an accurate description of the succession rules of the corresponding generating tree or an ansatz. By using this approach, we determine the generating trees for the pattern classes $I_n(000, 021)$, $I_n(100, 021)$, $I_n(110, 021)$, $I_n(102, 021)$, $I_n(100, 012)$, $I_n(011, 201)$, $I_n(011, 210)$ and $I_n(120, 210)$. Then we use the kernel method, obtain generating functions of each class, and find enumerating formulas. Lin and Yan studied the classification of the Wilf-equivalences for inversion sequences avoiding pairs of length-three patterns and showed that there are 48 Wilf classes among 78 pairs. In this paper, we solve six open cases for such pattern classes. Moreover, we extend the algorithm to restricted growth sequences and apply it to several classes. In particular, we present explicit formulas for the generating functions of the restricted growth sequences that avoid either $\{12313, 12323\}$, $\{12313, 12323, 12333\}$, or $\{123 \cdots \ell 1\}$.

© 2023 Elsevier Ltd. All rights reserved.

E-mail addresses: ikotsire@wlu.ca (I. Kotsireas), tmansour@univ.haifa.ac.il (T. Mansour), gokhan.yildirim@bilkent.edu.tr (G. Yıldırım).

¹ G. Yıldırım was partially supported by Scientific and Technological Research Council of Türkiye (TÜBİTAK)-Ardeb 120F352.

<https://doi.org/10.1016/j.jsc.2023.102231>

0747-7171/© 2023 Elsevier Ltd. All rights reserved.

1. Introduction

An *inversion sequence* of length n is an integer sequence $e = e_0 e_1 \cdots e_n$ such that $0 \leq e_i \leq i$ for each $0 \leq i \leq n$. We denote by I_n the set of inversion sequences of length n . There is a bijection between I_n and S_{n+1} , the set of permutations of length $n + 1$. Given any word τ of length k over the alphabet $[k] := \{0, 1, \dots, k - 1\}$, we say that an inversion sequence $e \in I_n$ contains the pattern τ if there is a subsequence of length k in e that is order isomorphic to τ ; otherwise, we say that e avoids the pattern τ . For instance, $e = 010213211 \in I_8$ avoids the pattern 201 because there is no subsequence $e_j e_k e_l$ of length three in e with $j < k < l$ and $e_k < e_l < e_j$. On the other hand, $e = 010213211$ contains the patterns 120 and 0000 because it has the subsequence $-- -2 -3 -1 -$ order isomorphic to 120, and the subsequence $-1 - -1 - -11$ order isomorphic to 0000. For a given pattern τ , we use $I_n(\tau)$ to denote the set of all τ -avoiding inversion sequences of length n . Similarly, for a given set of patterns B , we set $I_n(B) = \bigcap_{\tau \in B} I_n(\tau)$. Pattern-avoiding permutation classes have been thoroughly studied by researchers for more than forty years; for some highlights of the results, see (Kitaev, 2011) and references therein. A systematic study of pattern-avoidance for inversion sequences was initiated recently by Mansour and Shattuck (2015) for the patterns of length three with non-repeating letters and by Corteel et al. (2016) for repeating and non-repeating letters. Martinez and Savage (2018) generalized and extended the notion of pattern-avoidance for the inversion sequences to triples of binary relations that lead to new conjectures and open problems. Many successfully studied research programs for permutations such as pattern-avoidance in terms of vincular patterns, pairs of patterns, and longer patterns have already been initiated to study for inversion sequences; for some recent results, see (Auli and Elizalde, 2021; Beaton et al., 2019; Bouvel et al., 2018; Cao et al., 2019; Chern, 2023; Duncan and Steingrímsson, 2011; Hong and Li, 2022; Lin, 2018, 2020; Lin and Fu, 2021; Lin and Yan, 2020; Mansour and Shattuck, 2022; Yan and Lin, 2020) and references therein. In the context of inversion sequences, two sets of patterns B_1 and B_2 are considered Wilf equivalent if $|I_n(B_1)| = |I_n(B_2)|$ for all $n \geq 0$, that is, they have the same counting sequence. Note that there are thirteen patterns of length three up to order isomorphism; we denote them by $\mathcal{P}_3 = \{000, 001, 010, 100, 011, 101, 110, 021, 012, 102, 120, 201, 210\}$. Yan and Lin (2020) completed the classification of the Wilf-equivalences for inversion sequences avoiding pairs of length-three patterns. They showed that there are 48 Wilf classes among 78 pairs; for a complete list of the classes with open cases in terms of enumeration, see Table 1 and 2 in Yan and Lin (2020). In this paper, we solve six open cases for such pattern classes: $I_n(000, 021)$, $I_n(102, 021)$, $I_n(100, 012)$, $I_n(120, 210)$, Wilf-equivalent $I_n(011, 201)$ and $I_n(011, 210)$, and Wilf-equivalent $I_n(100, 021)$ and $I_n(110, 021)$. Recently, Testart (2022) also solved the following cases: $I_n(010, 000)$, $I_n(010, 110)$, $I_n(010, 120)$, and the Wilf-equivalent pairs $I_n(010, 201)$ and $I_n(010, 210)$.

For simplicity of the notation, we leave curly brackets and use $I_n(\tau_1, \dots, \tau_m)$ instead of $I_n(\{\tau_1, \dots, \tau_m\})$ for a given set of patterns $B = \{\tau_1, \dots, \tau_m\}$ throughout the paper. We shall use an algorithmic approach based on generating trees to enumerate pattern-restricted inversion sequences. For some earlier results, in the context of pattern-restricted permutations, see (Vatter, 2008; Zeilberger, 1998) and references therein. In this paper, we present applications of our algorithm only for the class $I_n(B)$ where either B includes a single pattern or a pair of patterns of length three. However, the method applies to other inversion sequences with various pattern restrictions; for an application of the method to a pattern of length four, see Mansour (2023). As we will see, the algorithm outputs either an accurate description of the succession rules of the generating tree for the given avoidance class or an ansatz based on which we can figure out the complete description of the generating tree. For most cases, we can use the kernel method (Banderier et al., 2002) to compute the generating functions and then obtain an exact enumerating formula for the corresponding pattern class or get a functional equation for the generating function. The latter case yields a procedure to calculate the coefficients of the generating function up to a given index.

We organize the paper as follows: In Section 2, we present our algorithm and demonstrate how it works on some examples such as $B = \{000, 001, 012\}$ and $B = \{000, 001\}$. In Section 3, we consider the open cases from single pattern of length three and obtain functional equations for the generating functions of $I_n(100)$, and Wilf-equivalent $I_n(201)$ and $I_n(210)$. In Section 4, we obtain the generating trees for the classes $I_n(000, 021)$, $I_n(100, 021)$, $I_n(110, 021)$, $I_n(102, 021)$, $I_n(100, 012)$,

$I_n(011, 201)$, $I_n(011, 210)$ and $I_n(120, 210)$ by using our algorithm. Then we use the kernel method, obtain the corresponding generating functions, and determine the counting sequences for them. In the last section, we extend our algorithm to the restricted growth sequences; see the last section of the paper for definitions. We present explicit formulas for the generating functions for the number of restricted growth sequences of length n that avoid either $\{12313, 12323\}$, $\{12313, 12323, 12333\}$, or $\{123 \dots \ell 1\}$.

2. An algorithm based on generating trees

Any set \mathcal{C} of discrete objects with a notion of a size such that for each n , there are finitely many objects of size n is called a combinatorial class. A *generating tree* (see West, 1996) for \mathcal{C} is a rooted, labelled tree whose vertices are the objects of \mathcal{C} with the following properties: (i) each object of \mathcal{C} appears exactly once in the tree; (ii) objects of size n appear at level n in the tree (the root has level 0); (iii) the children of some object are obtained by a set of succession rules of the form that determines the number of children and their labels.

Note that any pattern over the alphabet $[k]$ can be extended to an inversion sequence. Suppose a pattern $\tau = \tau_1 \dots \tau_m$ is given and let $\{0, 1, \dots, t\}$ denote the set of all letters appeared in τ . We define L_τ to be the set of all inversion sequences $\theta^{(1)}\tau_1\theta^{(2)}\tau_2 \dots \theta^{(m)}\tau_m$ such that the length of the inversion sequence $\theta^{(1)}\tau_1\theta^{(2)}\tau_2 \dots \theta^{(j)}\tau_j$ is minimal for each $j = 1, 2, \dots, m$. Note that some words $\theta^{(j)}$ s might be empty. By the minimality condition on the lengths of $\theta^{(1)}, \dots, \theta^{(m)}$, we have that the length of any pattern in L_τ is at most $m + t$. For instance, if $\tau = 021$, then $m = 3, t = 2$, and $L_\tau = \{0021, 0121\}$; if $\tau = 001$, then $m = 3, t = 1$, and $L_\tau = \{001\}$. Clearly, any inversion sequence e avoids B if and only if e avoids $L = \cup_{\{\tau \in B\}} L_\tau$. For any set of patterns B , we identify B with the set of patterns $L_B = \cup_{\{\tau \in B\}} L_\tau$.

For a given set of patterns B , let $\mathcal{I}_B = \cup_{n=0}^\infty I_n(B)$. We will construct a pattern-avoidance tree $\mathcal{T}(B)$ for the class of pattern-avoiding inversion sequences \mathcal{I}_B . The tree $\mathcal{T}(B)$ is considered empty if no inversion sequence of arbitrary length avoids the set B . Otherwise, the root can always be taken as 0, that is, $0 \in \mathcal{T}(B)$. Starting with this root which stays at level 0, the remainder of the tree $\mathcal{T}(B)$ can then be constructed in a recursive manner such that the n^{th} level of the tree consists of exactly the elements of $I_n(B)$ arranged in such a way that the parent of an inversion sequence $e_0e_1 \dots e_n \in I_n(B)$ is the unique inversion sequence $e_0e_1 \dots e_{n-1} \in I_{n-1}(B)$. The children of $e_0e_1 \dots e_{n-1} \in I_{n-1}(B)$ are obtained from the set $\{e_0e_1 \dots e_{n-1}e_n \mid e_n = 0, 1, \dots, n\}$ by applying the pattern restrictions of the patterns in B . We arrange the nodes from the left to the right so that if $e = e_0e_1 \dots e_{n-1}i$ and $e' = e_0e_1 \dots e_{n-1}j$ are children of the same parent $e_1 \dots e_{n-1}$, then e appears on the left of e' if $i < j$. See Fig. 1 for the first few levels of $\mathcal{T}(\{012\})$. Note that the size of $I_n(B)$ equals the number of nodes in the n -th level of $\mathcal{T}(B)$.

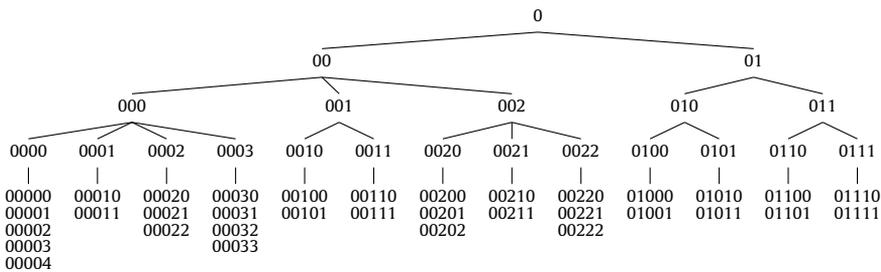


Fig. 1. First four levels of $\mathcal{T}(\{012\})$.

For a given set of patterns B , it plays an essential role to understand the nature of the tree $\mathcal{T}(B)$ to enumerate the class $\mathcal{I}_B = \cup_{n=0}^\infty I_n(B)$. Let $\mathcal{T}(B; e)$ denote the subtree consisting of the inversion sequence e as the root and its descendants in $\mathcal{T}(B)$. In our arguments, it will be important to determine if the subtrees starting from two distinct nodes $e, e' \in \mathcal{T}(B)$ are isomorphic or not, that is, $\mathcal{T}(B; e) \cong \mathcal{T}(B; e')$ in the sense of plane tree isomorphism. Lemma 2.1 provides an easy to check criteria for this task, for similar results, see Brändén and Mansour (2005).

Lemma 2.1. Let t be the length of the longest pattern in B . We have that $\mathcal{T}(B; e) \cong \mathcal{T}(B; e')$ for two inversion sequences $e, e' \in \mathcal{T}(B)$ if and only if $\mathcal{T}^{2t}(B; e) \cong \mathcal{T}^{2t}(B; e')$ where $\mathcal{T}^m(B; e)$ denotes the finite tree corresponding to the first $m - 1$ level of $\mathcal{T}(B; e)$.

Proof. Since avoiding B is equivalent to avoiding $L_B = \cup_{\{\tau \in B\}} L_\tau$ in the set of inversion sequences, we assume that any pattern in B is an inversion sequence.

Let $e, e' \in \mathcal{T}(B)$. Clearly, $\mathcal{T}(B; e) \cong \mathcal{T}(B; e')$ implies $\mathcal{T}^{2t}(B; e) \cong \mathcal{T}^{2t}(B; e')$.

Now, let us assume that $\mathcal{T}(B; e) \not\cong \mathcal{T}(B; e')$ as plane trees. We read the nodes of $\mathcal{T}(B; e)$ (resp. $\mathcal{T}(B; e')$) from top to bottom and from left to right and denote them as e_j (resp. e'_j) with $e_0 = e$ (resp. $e'_0 = e'$). Since $\mathcal{T}(B; e) \not\cong \mathcal{T}(B; e')$, there exists $s \geq 0$ minimal such that (1) the number children of e_j equals the number of children of e'_j , for $j = 1, 2, \dots, s - 1$, and (2) the number of children of e_s does not equal the number of children of e'_s . By construction of $\mathcal{T}(B)$, for all $j = 1, 2, \dots, s - 1$, there exist letters p_{ij}, q_{ij} such that the inversion sequences $ef_j := ep_{1j}p_{2j} \cdots p_{ijj}$ and $e'f'_j := e'q_{1j}q_{2j} \cdots q_{ijj}$ avoid B , while the inversion sequence $ef_s := ep_1p_2 \cdots p_{i_s}$ contains $\tau \in B$; the inversion sequence $e'f'_s = e'q_1q_2 \cdots q_{i_s}$ avoids B and there exists a bijection α such that $q_j = \alpha(p_j)$, for all $j = 1, 2, \dots, i_s$.

Any occurrence of τ in ef_s can use at most $t - 1$ letters of f_s . Thus, there is a subsequence $g = p_{k_1} \cdots p_{k_m}$ of f_s of minimal length m such that the word eg contains τ and $m \leq t - 1$ but the word $e'g'$ avoids B , where $g' = q_{k_1} \cdots q_{k_{t-1}}$ is a subsequence of f'_s .

Since ef_s is an inversion sequence, then there exists an inversion sequence $e\tilde{g} \in L_{eg}$ such that $e\tilde{g}$ is a subsequence of ef_s . Since each letter p_j is mapped to the letter q_j by α , we see that the sequence \tilde{g} is mapped to \tilde{g}' . Since \tilde{g}' is a subsequence of f'_s , we see that $e'\tilde{g}'$ is an inversion sequence in $L_{e'g'}$.

Thus, the inversion sequence $e\tilde{g}$ contains τ such that the length of \tilde{g} is at most $t + t - 1$ and the inversion sequence $e'\tilde{g}'$ avoids τ such that the length of \tilde{g}' is at most $t + t - 1$. Hence, $\mathcal{T}^{2t}(B; e) \not\cong \mathcal{T}^{2t}(B; e')$. \square

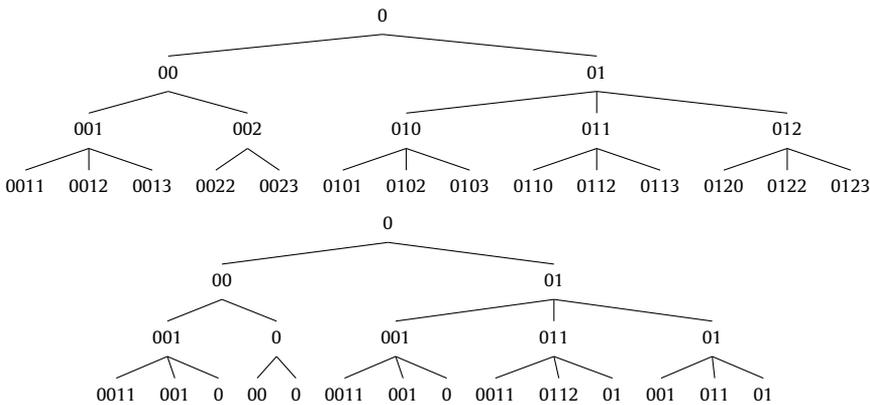


Fig. 2. First three levels of $\mathcal{T}(\{000, 021\})$ and $\mathcal{T}[\{000, 021\}]$.

We define an equivalence relation on the set of nodes of $\mathcal{T}(B)$ as follows. Let $v = v_0v_1 \cdots v_a$ and $w = w_0w_1 \cdots w_b$ be two nodes in $\mathcal{T}(B)$. We say that v is equivalent to w , denoted by $v \sim w$, if and only if $\mathcal{T}(B; v) \cong \mathcal{T}(B; w)$. Note that Lemma 2.1 performs a finite procedure for checking $v \sim w$. Define $V[B]$ to be the set of all equivalence classes in the quotient set $\mathcal{T}(B)/\sim$. We will represent each equivalence class $[v]$ by the label of the unique node v which appears on the tree $\mathcal{T}(B)$ as the left-most node at the lowest level among all other nodes in the same equivalence class. Let $\mathcal{T}[B]$ be the same tree $\mathcal{T}(B)$ where we replace each node v by its equivalence class label, see Fig. 2. That is, w is relabelled by v such that

- $v \sim w$, and

- either $a < b$ or $a = b$ such that, in the list of the nodes at level a in the tree $\mathcal{T}(B)$ from left to right, the node v does appear before the node w .

Next, we define an algorithm for finding $\mathcal{T}[B]$ for a given set of patterns B with $0 \notin B$. The algorithm takes the pattern set B and an integer D as input and outputs a set of succession rules for the corresponding generating tree. As we run the algorithm, we use the set Q_D and R to keep track of the equivalence classes that are obtained at set D and the deduced succession rules for the generating tree that are obtained up to step D , respectively. The details are as follows:

- (1) We **initialize** the tree $\mathcal{T}[B]$ by the root 0, and define $Q_0 = \{0\}$ and $R = \emptyset$.
- (2) Let D be any positive integer.
- (3) For all $i = 1, 2, \dots, D$,
 - (3.1) for any $w \in Q_{i-1}$, we denote the set of all children of w in $\mathcal{T}(B)$ by N_w . We **denote** the set of all children of all new equivalence classes at i^{th} step by $M_i = \cup_{w \in Q_{i-1}} N_w$. If $M_i = \emptyset$, then we stop the loop and go to (4).
 - (3.2) we **initialize** the set Q_i (set of new equivalence classes at i^{th} step) to be empty set. For each child w in M_i ,
 - (3.2.1) we find $v \in \cup_{j=0}^{i-1} Q_j$, if possible, such that $w \sim v$, where we use Lemma 2.1 to check that $w \sim v$ holds or not;
 - (3.2.2) otherwise, we add the equivalence class w to Q_i .
 - (3.3) based on (3.2), we **add** the rule $w \rightsquigarrow v_1 v_2 \dots v_s$ to the set R , where v_j is the label of the j^{th} child of w , from left to right, in $\mathcal{T}[B]$.
- (4) If we **stop** at (3.1), then we have the finite set of labels $\cup_{j=0}^{i-1} Q_j$ and finite set of succession rules R that specifies the tree $\mathcal{T}[B]$ with the root 0. In this context, B is called *regular*.
- (5) Otherwise, we have set of succession rules R that specifies the tree $\mathcal{T}[B]$ with its root 0 up to level $k(D)$ where $k(D)$ is an integer depending on D . We could **guess**, if possible, all the set of succession rules of $\mathcal{T}[B]$ based on R , then use Lemma 2.1 to prove this claim. In case we fail to guess the whole set of the succession rules, then either we increase D or we say that our procedure does not lead us to determine all succession rules of $\mathcal{T}[B]$.

We will use the following fact throughout the paper: for any pattern collection B , $\mathcal{T}(B) \cong \mathcal{T}[B]$ (as plane trees) and the number of nodes at the n^{th} level of the generating tree is equal to the number of inversion sequences of length n avoiding the patterns in B .

Example 2.2. Let $B = \{000, 001, 012\}$, we apply our procedure with $D = 5$ as follows:

i	M_i	Comments	Q_i	R
0			{0}	\emptyset
1	{00, 01}	$00 \approx 0, 01 \approx 0, 01 \approx 00$	{00, 01}	$\{0 \rightsquigarrow 00, 01\}$
2	{010, 011}	$010 \approx 00, 011 \approx v \in Q_0 \cup Q_1$	{011}	$\{0 \rightsquigarrow 00, 01, 01 \rightsquigarrow 00, 011\}$
3	{0110}	$0110 \sim 00$	\emptyset	$\{0 \rightsquigarrow 00, 01, 01 \rightsquigarrow 00, 011, 011 \rightsquigarrow 00\}$.

Hence, the generating tree $\mathcal{T}[B]$ given by the algorithm has the following succession rules:

Root: 0, Rules: $0 \rightsquigarrow 00, 01, 01 \rightsquigarrow 00, 011, 011 \rightsquigarrow 00$.

We want to find the generating function $R(x) = \sum_{n \geq 0} |I_n(B)|x^{n+1}$. We use $A_w(x)$ to denote the generating function for the number of nodes in the subtree $\mathcal{T}(B; w)$. Hence, by the generating tree $\mathcal{T}[B]$, we have $R(x) = x + xA_{00}(x) + xA_{01}(x)$, $A_{00}(x) = x$, $A_{01}(x) = x + xA_{00}(x) + xA_{011}(x)$, and $A_{011}(x) = x + xA_{00}(x)$. By solving for $R(x)$, we obtain that $R(x) = x^4 + 2x^3 + 2x^2 + x$.

Example 2.3. Let $B = \{000, 001\}$. By applying our procedure with $D = 5$, we guess that the tree $\mathcal{T}[B]$ is given by

$$\text{Root: } a_0, \text{ Rules: } a_0 \rightsquigarrow b_0a_1, \quad a_m \rightsquigarrow b_0b_1b_2 \dots b_ma_{m+1} \quad b_m \rightsquigarrow b_0b_1b_2 \dots b_{m-1},$$

where $a_0 = 0$, $a_m = 012 \dots m$, $b_0 = 00$ and $b_m = 012 \dots (m - 1)mm$ for $m \geq 1$. We will make use of Lemma 2.1 to verify the succession rules of the generating tree. Since other cases are very similar, we only show that the succession rule $a_m \rightsquigarrow b_0b_1b_2 \dots b_ma_{m+1}$ holds. Let $v = 012 \dots m$, then the children of v in $\mathcal{T}(B)$ are $012 \dots mj$ with $j = 0, 1, \dots, m + 1$. By using Lemma 2.1, we see that $012 \dots m0 \sim 00$, $012 \dots mj \sim 012 \dots (j - 1)jj$ with $j = 1, 2, \dots, m$, and for $j = m + 1$ we have a new equivalence class $012 \dots m(m + 1)$. This verifies the succession rule $a_m \rightsquigarrow b_0b_1b_2 \dots b_ma_{m+1}$.

We aim at computing the generating function $R(x) = \sum_{n \geq 0} |I_n(B)|x^{n+1}$. We use $A_w(x)$ to denote the generating function for the number of nodes in the subtree $\mathcal{T}(B; w)$. Let us define $B_m(x) = A_{012 \dots m}(x)$ and $C_m(x) = A_{012 \dots (m-1)mm}(x)$, for $m \geq 1$. Then, by the generating tree $\mathcal{T}[B]$, we obtain that $R(x) = x + xA_{00}(x) + xA_{01}(x)$, $A_{00}(x) = x$, and

$$B_m(x) = x + x^2 + x(C_1(x) + \dots + C_m(x)) + xB_{m+1}(x),$$

$$C_m(x) = x + x^2 + x(C_1(x) + \dots + C_{m-1}(x)).$$

We define $G(x, u) = \sum_{m \geq 1} G_m(x)u^{m-1}$, where $G \in \{B, C\}$. Hence, by multiplying the recurrence relations by u^{m-1} and summing over $m \geq 1$, we have

$$B(x, u) = \frac{x(1+x)}{1-u} + \frac{x}{1-u}C(x, u) + \frac{x}{u}(B(x, u) - B(x, 0)), \tag{2.1}$$

$$C(x, u) = \frac{x(1+x)}{1-u} + \frac{x}{1-u}C(x, u) - xC(x, u). \tag{2.2}$$

By solving (2.2) for $C(x, u)$, we have

$$C(x, u) = \frac{x(1+x)}{(1+x)(1-u) - x}.$$

The equations of type (2.1) can be solved systematically using the kernel method (Banderier et al., 2002). In this case, if we assume that $u = x$, then

$$B(x, 0) = \frac{x(1+x)}{1-x} + \frac{x}{1-x}C(x, x) = \frac{x(1+x)}{1-x-x^2}. \tag{2.3}$$

Hence, by comparing coefficients of x^{n+1} , we obtain that $|I_n(000, 001)| = \text{Fib}_{n+2}$, where Fib_n is the n^{th} Fibonacci number, that is, $\text{Fib}_n = \text{Fib}_{n-1} + \text{Fib}_{n-2}$ with $\text{Fib}_0 = 0$ and $\text{Fib}_1 = 1$.

3. Set of patterns $B \subset \mathcal{P}_3$ with $|B| = 1$

As we discussed in the introduction, the first systematic study of pattern-avoiding inversion sequences was carried out for the case of a single pattern of length three in Corteel et al. (2016) and Mansour and Shattuck (2015). The results of these papers demonstrated that there are some remarkable connections with pattern-restricted inversion sequences and other well-studied combinatorial structures. Some of the highlights of their results can be summarized as follows: the odd-indexed Fibonacci numbers count $I_n(012)$, the large Schröder numbers count $I_n(021)$, the Euler up/down numbers count $I_n(000)$, the Bell numbers count $I_n(011)$, and powers of two count $I_n(001)$; for details see the above references. There are still no enumerating formulas for the avoidance sets $I_n(100)$ and $I_n(120)$, and Wilf-equivalent $I_n(201)$ and $I_n(210)$. For the enumeration of the pattern 010, see the recent preprint (Testart, 2022). In this section, we use this paper’s algorithmic approach and derive functional equations for the generating functions of the classes $I_n(100)$ and $I_n(201)$. For similar results in the context of pattern-restricted permutations, see Nakamura and Zeilberger (2013), Yang and Zeilberger (2020).

3.1. Class $I_n(100)$

From now, we denote the constant word $kk \cdots k$ of length d by k^d , for any letter k and positive integer d . Our algorithm allows us to guess the generating tree $\mathcal{T}[\{100\}]$.

Theorem 3.1. *The generating tree $\mathcal{T}[\{100\}]$ is given by*

$$\text{Root: } a_1, \quad \text{Rules: } a_m \rightsquigarrow a_{m+1}b_{m,1} \cdots b_{m,m}, \quad b_{m,j} \rightsquigarrow (b_{m,j-1})^j b_{m+1,j} \cdots b_{m+1,m+1},$$

where $a_m = 0^m$ and $b_{m,j} = 0^m j$, for all $1 \leq j \leq m$.

Proof. We proceed by using our algorithm. We label the inversion sequences $0 \in I_0$ by 0. Clearly, the children of 0^m are $0^{m+1}, 0^m 1, \dots, 0^m m$. Thus it remains to show that the children of $0^m j$ are $(0^m(j-1))^j (0^{m+1} j) \cdots (0^{m+1}(m+1))$. Let $v_i = 0^m j i$, we have that

- if $i = 0, 1, \dots, j-1$, then $v_i \sim 0^m(j-1)$ in $\mathcal{T}(\{100\})$ (by removing the letter i and subtracting 1 from each letter bigger than i);
- if $i = j, j+1, \dots, m$, then $v_i = 0^m j i \sim 0^{m+1} i$ in $\mathcal{T}(\{100\})$ (by replacing the letter j by 0),

which completes the proof. \square

To study the generating function $R(x) = \sum_{n \geq 0} |I_n(100)|x^{n+1}$, we define $A_m(x)$ and $B_{m,j}(x)$ to be the generating functions for the number of nodes in the subtrees $\mathcal{T}(B; 0^m)$ and $\mathcal{T}(B; 0^m j)$, respectively. Let $B_m(x) = \sum_{j=1}^m B_{m,j}(x)$ and $B_{m,0}(x) = A_{m+1}(x)$. Thus, from the generating tree's succession rules, we get

$$A_m(x) = x + xA_{m+1}(x) + xB_m(x), \quad m \geq 1,$$

$$B_{m,j}(x) = x + jxB_{m,j-1}(x) + xB_{m+1,j}(x) + \cdots + xB_{m+1,m+1}(x), \quad j = 1, 2, \dots, m.$$

Then, we define the following bivariate generating functions: $A(x, v) = \sum_{m \geq 1} A_m(x)v^{m-1}$, $B_m(x, u) = \sum_{j=1}^m B_{m,j}(x)u^{m-j}$, and $B(x, v, u) = \sum_{m \geq 1} B_m(x, u)v^{m-1}$. Note that the system of recurrences can be written as follows:

$$A(x, v) = \frac{x}{1-v} + \frac{x}{v}(A(x, v) - A(x, 0)) + xB(x, v, 1),$$

$$B(x, v, u) = \frac{x}{(1-v)(1-vu)} - xu \frac{\partial}{\partial u} \frac{B(x, v, u) - B(x, v, 0)}{u} - xu \frac{\partial}{\partial u} \frac{A(x, uv) - A(x, 0)}{uv}$$

$$+ x \frac{\partial}{\partial v} \left(\frac{v}{u} (B(x, v, u) - B(x, v, 0)) + \frac{A(x, uv) - A(x, 0)}{u} \right)$$

$$+ \frac{x}{uv(1-u)} (B(x, v, u) - uB(x, uv, 1) - (1-u)B(x, 0, 0))$$

$$- \frac{x}{uv} (B(x, v, 0) - B(x, 0, 0)).$$

By taking $v = x$ into the first equation, we get the following result.

Theorem 3.2. *The generating function $\sum_{n \geq 0} |I_n(100)|x^{n+1}$ is equal to $A(x, 0)$ that satisfies the following functional equation:*

$$A(x, 0) = \frac{x}{1-x} + xB(x, x, 1).$$

We can not solve this functional equation to obtain an explicit expression for the generating function. But the functional equation can be used to obtain the first n terms of the generating function $A(x, 0)$ for any positive integer n . The first 24 terms are 1, 2, 6, 23, 106, 565, 3399, 22678, 165646, 1311334, 11161529, 101478038, 980157177, 10011461983, 107712637346, 1216525155129, 14380174353934, 177440071258827, 2280166654498540, 30450785320307436, 421820687108853017, 6050801956624661417, 89738550379292147192, 1374073440225390131037, 21694040050913295537753.

3.2. Class $I_n(201)$ or $I_n(210)$

Based on the algorithm’s ansatz, we get the same succession rules for the generating trees of $I_n(201)$ and $I_n(210)$. The generating tree is given as follows (from now on we state the generating trees without proofs, since they can be handled similarly to the proof of Theorem 3.1):

$$\begin{aligned} \text{Root: } a_1, \quad \text{Rules: } a_m \rightsquigarrow a_{m+1}a_{m+1}b_{m,2}b_{m,3}\cdots b_{m,m}, \\ b_{m,j} \rightsquigarrow a_{m+3-j}b_{m+3-j,2}\cdots b_{m+1,j}b_{m+1,j+1}\cdots b_{m+1,m+1}, \end{aligned}$$

where $a_m = 0^m$ and $b_{m,j} = 0^m j$, for all $m \geq 1$ and $2 \leq j \leq m$.

This result implies the following corollary:

Corollary 3.3. $|I_n(201)| = |I_n(210)|$ for all $n \geq 1$.

For a bijection between these two classes, see Mansour and Shattuck (2015).

To study the generating function $R(x) = \sum_{n \geq 0} |I_n(201)|x^{n+1}$, we define $A_m(x)$ and $B_{m,j}(x)$ to be the generating functions for the number of nodes in the subtrees $\mathcal{T}(B; 0^m)$ and $\mathcal{T}(B; 0^m j)$, respectively. Let $B_m(x) = \sum_{j=2}^m B_{m,j}(x)$. Thus,

$$A_m(x) = x + 2xA_{m+1}(x) + xB_m(x), \quad m \geq 1, \tag{3.1}$$

$$B_{m,j}(x) = x + xA_{m+3-j}(x) + x \sum_{i=2}^j B_{m+1-j+i,i}(x) + x \sum_{i=j}^{m+1} B_{m+1,i}(x), \quad 1 \leq j \leq m. \tag{3.2}$$

Clearly, $A_1(x) = x + 2xA_2(x)$.

We define the generating functions: $A(x, v) = \sum_{m \geq 1} A_m(x)v^{m-1}$, $B_m(x, u) = \sum_{j=2}^m B_{m,j}(x)u^{m-j}$, and $B(x, v, u) = \sum_{m \geq 2} B_m(x, u)v^{m-2}$. Then by multiplying (3.1) by v^{m-1} and summing over $m \geq 1$, we obtain

$$A(x, v) = \frac{x}{1-v} + \frac{2x}{v}(A(x, v) - A(x, 0)) + xvB(x, v, 1). \tag{3.3}$$

Note that $A_2(x) = (A(x, 0) - x)/(2x)$.

By multiplying (3.2) by $u^{m-j}v^{m-1}$ and summing over $2 \leq j \leq m$, we obtain

$$\begin{aligned} B(x, v, u) &= \frac{x}{(1-v)(1-vu)} + \frac{x}{u^2v^2(1-v)}(A(x, uv) - \frac{uv}{2x}(A(x, 0) - x) - A(x, 0)) \\ &+ \frac{x}{uv(1-v)}(B(x, v, u) - B(x, v, 0)) \\ &+ \frac{x}{uv(1-u)}(B(x, v, u) - uB(x, uv, 1)) - \frac{x}{uv}B(x, v, 0). \end{aligned}$$

Hence, by setting $v = 2x$ into (3.3), we obtain the following functional equation for the generating function. For a similar functional equation, see Mansour and Shattuck (2015).

Theorem 3.4. *The generating function $\sum_{n \geq 0} |I_n(201)|x^{n+1}$ is equal to $A(x, 0)$ that satisfies the following functional equation:*

$$A(x, 0) = \frac{x}{1 - 2x} + 2x^2B(x, 2x, 1).$$

By using the above theorem, we can obtain the first n terms of the generating function $A(x, 0)$ for any positive integer n . The first 24 terms are 1, 2, 6, 24, 118, 674, 4306, 29990, 223668, 1763468, 14558588, 124938648, 1108243002, 10115202962, 94652608690, 905339525594, 8829466579404, 87618933380020, 883153699606024, 9028070631668540, 93478132393544988, 979246950529815364, 10368459385853924212, 110866577818487410864.

4. Set of patterns $B \subset \mathcal{P}_3$ with $|B| = 2$

Inversion sequences avoiding pairs of patterns of length three was first systematically studied by Yan and Lin (2020). They determined that there are 48 Wilf classes among 78 pairs and provided enumerating formulas for some of the classes; for a complete list see Table 1 and 2 in Yan and Lin (2020). In this section, we first obtain the generating trees corresponding to the classes $I_n(000, 021)$, $I_n(100, 021)$, $I_n(110, 021)$, $I_n(102, 021)$, $I_n(100, 012)$, $I_n(011, 201)$, $I_n(011, 210)$ and $I_n(120, 210)$ by using our algorithm. It will follow from the generating trees that classes $I_n(011, 201)$ and $I_n(011, 210)$ are Wilf-equivalent, and $I_n(100, 021)$ and $I_n(110, 021)$ are Wilf-equivalent. Then we use the kernel method and determine the counting sequences for them, see Table 1. For some additional new results in this direction, see Testart (2022).

Table 1
Summary of the results.

B	$a_n = I_n(B) $	reference
(000,021)	$\frac{1}{2}(3a_{n-1} + a_n - 3a_{n+1} + a_{n+2})$ $a_n = \sum_{k=0}^n (-1)^{n-k} \binom{n}{k} \binom{2k}{k}$	Theorem 4.1
(100,021)~(110,021)	$\frac{n^2+n+6}{2(n+3)(n+2)} \binom{2n+2}{n+1}$	Theorem 4.3
(102,021)	$\sum_{k=0}^n \frac{1}{k+1} \binom{2k}{k} - 1 - \frac{1}{6}n^3 - \frac{11}{6}n + 2^n$	Theorem 4.4
(100,012)	$\frac{(n+7)Fib_n + 15Fib_{n+1} + nFib_{n+2}}{5} - 1 - \binom{n+2}{2}$	Theorem 4.10
(011,201)~(011,210)	functional equation for the generating function	Theorem 4.11
(120,210)	functional equation for the generating function	Theorem 4.12

4.1. Class $I_n(000, 021)$

Let $B = \{000, 021\}$. When we apply our algorithm to the pattern class $I_n(B)$, we obtain a generating tree that leads to an enumerating formula for this case.

We define $r_0 = 0$, $a_m = 0011 \dots mm$, $b_m = 0011 \dots (m - 1)(m - 1)m$ with $m \geq 0$, and $c_m = 01122 \dots mm$, $d_m = 01122 \dots (m - 1)(m - 1)m$ with $m \geq 1$. The generating tree $\mathcal{T}[B]$ is given by

$$\begin{aligned} \text{Root: } r_0, \text{ Rules: } r_0 \rightsquigarrow a_0d_1, \quad a_m \rightsquigarrow b_{m+1}b_m \dots b_0, \quad b_m \rightsquigarrow a_mb_mb_{m-1} \dots b_0, \quad m \geq 0, \\ c_m \rightsquigarrow a_md_{m+1}d_m \dots d_1, \quad d_m \rightsquigarrow b_mc_md_md_{m-1} \dots d_1, \quad m \geq 1. \end{aligned}$$

This result follows from the following observations. We label the inversion sequences $0 \in I_0$ and $00, 01 \in I_1$ by r_0 and a_0, d_1 , respectively. Thus, $r_0 \rightsquigarrow a_0d_1$. It remains to show that the generating tree's succession rules hold. Since the other cases are very similar, we will verify only the rule $a_m \rightsquigarrow b_{m+1}b_m \dots b_0$ for all $m \geq 0$. Let $e = e_0e_1 \dots e_n$ be any inversion sequence is labelled by a_m . So, by definitions, we have that $\mathcal{T}(B; e) \cong \mathcal{T}(B; a_m)$. On the other hand, the inversion sequence that labelled by $a_mj = 0011 \dots mmj$ where $j = m + 1, m + 2, \dots, 2m + 2$ (otherwise, a_mj does not avoid B). Moreover, (i) $a_m(m + 1) = 0011 \dots mm(m + 1) = b_{m+1}$; (ii) $a_m(m + j) = 0011 \dots mm(m + j)$;

the subtree $\mathcal{T}(B; a_m(m + j))$ is isomorphic to the subtree $\mathcal{T}(B; b_{m+2-j})$ by removing the letters $m + 2 - j, m + 3 - j, \dots, m$ and decreasing each letter greater than m by $2j - 1$. Thus, Lemma 2.1 gives the children of the node with label a_m are exactly the nodes labelled by b_{m+1}, b_m, \dots, b_0 , that is, $a_m \rightsquigarrow b_{m+1}b_m \cdots b_0$ with $m \geq 0$.

In order to find an explicit formula for the generating function for the number of inversion sequences in $I_n(B)$, we define $R(x)$ (respectively, $A_m(x)$, $B_m(x)$, $C_m(x)$, and $D_m(x)$) to be the generating function for the number of nodes in the subtrees $\mathcal{T}(B; 0)$ (respectively, $\mathcal{T}(B; a_m)$, $\mathcal{T}(B; b_m)$, $\mathcal{T}(B; c_m)$, and $\mathcal{T}(B; d_m)$), where its root is at level 0. Hence, by the rules of the tree $\mathcal{T}(B)$, we have

$$R(x) = x + xA_0(x) + xD_1(x), \tag{4.1}$$

$$A_m(x) = x + x \sum_{j=0}^{m+1} B_j(x), \quad m \geq 0, \tag{4.2}$$

$$B_m(x) = x + xA_m(x) + x \sum_{j=0}^m B_j(x), \quad m \geq 0, \tag{4.3}$$

$$C_m(x) = x + xA_m(x) + x \sum_{j=1}^{m+1} D_j(x), \quad m \geq 0, \tag{4.4}$$

$$D_m(x) = x + xB_m(x) + xC_m(x) + x \sum_{j=1}^m D_j(x), \quad m \geq 0. \tag{4.5}$$

We define $A(x, u) = \sum_{m \geq 0} A_m(x)u^m$, $B(x, u) = \sum_{m \geq 0} B_m(x)u^m$, $C(x, u) = \sum_{m \geq 1} C_m(x)u^{m-1}$, and $D(x, u) = \sum_{m \geq 1} D_m(x)u^{m-1}$. Hence, (4.1)-(4.5) can be written as

$$R(x) = x + xA(x, 0) + xD(x, 0), \tag{4.6}$$

$$A(x, u) = \frac{x}{1-u} + \frac{x}{u}(B(x, u) - B(x, 0)) + \frac{x}{1-u}B(x, u), \tag{4.7}$$

$$B(x, u) = \frac{x}{1-u} + xA(x, u) + \frac{x}{1-u}B(x, u), \tag{4.8}$$

$$C(x, u) = \frac{x}{1-u} + \frac{x}{u}(A(x, u) + D(x, u) - A(x, 0) - D(x, 0)) + \frac{x}{1-u}D(x, u), \tag{4.9}$$

$$D(x, u) = \frac{x}{1-u} + \frac{x}{u}(B(x, u) - B(x, 0)) + xC(x, u) + \frac{x}{1-u}D(x, u). \tag{4.10}$$

By (4.7)-(4.8), we have

$$\frac{(1-x)u - x^2 - u^2}{u(1-u-x)}A(x, u) = -\frac{x^2}{u(1-x)}A(x, 0) + \frac{x}{(1-u-x)(1-x)}. \tag{4.11}$$

In this case for the kernel method, if we assume that $u = x^2M(x)$, where $M(x) = \frac{1-x-\sqrt{1-2x-3x^2}}{2x^2}$ is the generating function for the Motzkin numbers, see Sloane (1996, Sequence A001006), then we obtain

$$A(x, 0) = \frac{xM(x)}{1-x-x^2M(x)} = xM^2(x).$$

By (4.11)

$$A(x, u) = \frac{x(ux + x^2 - x)A(x, 0) + u}{(1-x)(u(1-x) - x^2 - u^2)} = \frac{xM(x)(x^2M(x) - u)}{u^2 + (x-1)u + x^2}. \tag{4.12}$$

Thus, by (4.8)

$$B(x, u) = \frac{x(1 + (1 - u)A(x, u))}{1 - u - x} = \frac{xM(x)((u + x)x^2M(x) - u + x^2)}{u^2 + u(x - 1) + x^2}. \tag{4.13}$$

By substituting (4.9) into (4.10) with using (4.12)-(4.13), we obtain

$$\begin{aligned} & \frac{((1 - x)u - x^2 - u^2)^2}{u(1 - u)}D(x, u) + \frac{x^2((1 - x)u - x^2 - u^2)}{u}D(x, 0) \\ &= \frac{xM(x)((x^2 - x - 1)u^2 + (x^3 - 3x^2 + 1)u + x^4 + x^2)}{1 - u} \\ &+ \frac{x^3M^2(x)((x - 1)u^2 + (x^2 - x + 3)u + x^3 + x^2 + x - 2)}{1 - u}. \end{aligned} \tag{4.14}$$

By differentiating this equation with respect to u and taking $u = x^2M(x)$, after some simple algebraic simplifications, we get

$$D(x, 0) = \frac{x((x^2 + 2x - 2)M(x) - x + 1)((1 - x)M(x) - 2)}{(1 + x)(1 - 3x)}.$$

Hence, by (4.6), we obtain the following result.

Theorem 4.1. *The generating function $R(x) = \sum_{n \geq 0} |I_n(000, 021)|x^{n+1}$ is given by*

$$\begin{aligned} & \frac{3x^3 + x^2 - 3x + 1}{2x^2\sqrt{(1+x)(1-3x)}} - \frac{(1-x)^2}{2x^2} \\ &= x + 2x^2 + 5x^3 + 14x^4 + 39x^5 + 111x^6 + 317x^7 + 911x^8 + 2627x^9 + 7600x^{10} + \dots \end{aligned}$$

Moreover, by Sequence A002426 in Sloane (1996), we get for all $n \geq 1$,

$$|I_n(000, 021)| = \frac{1}{2}(3a_{n-1} + a_n - 3a_{n+1} + a_{n+2}),$$

where $a_n = \sum_{k=0}^n (-1)^{n-k} \binom{n}{k} \binom{2k}{k}$.

4.2. Class $I_n(100, 021)$ and $I_n(110, 021)$

In this section, we will provide the rules for the generating trees $\mathcal{T}[\{100, 021\}]$ and $\mathcal{T}[\{110, 021\}]$. The generating trees show that these two classes are equinumerous, and also lead to an exact enumerating formula. When we apply our algorithm to the class $I_n(100, 021)$, we obtain the following rules for $\mathcal{T}[\{100, 021\}]$:

$$\begin{aligned} \text{Root: } a_0, \text{ Rules: } & a_m \rightsquigarrow a_{m+1}b_m b_{m-1} \cdots b_1, \quad b_m \rightsquigarrow c_m b_{m+1} b_m \cdots b_1, \quad m \geq 1, \\ & c_m \rightsquigarrow c_{m+1}d_m d_{m-1} \cdots d_1 e, \quad d_m \rightsquigarrow d_{m+1}d_m \cdots d_1 e, \quad m \geq 1, \\ & e \rightsquigarrow d_1 e, \end{aligned}$$

where $a_m = 0^m 1$, $b_m = 0^m 1$, $c_m = 0^m 10$, $d_m = 0^m 102$ and $e = 0103$. From a very similar argument presented in Section 4.1, it follows that the number of nodes at level n (the root is at level 0) in $\mathcal{T}[\{100, 021\}]$ is equal to the number of inversion sequences in $I_n(100, 021)$.

Next, we will apply our algorithm to the class $I_n(110, 021)$, and obtain the rules for the generating tree $\mathcal{T}[\{110, 021\}]$:

$$\begin{aligned} \text{Root: } a_0, \text{ Rules: } & a_m \rightsquigarrow a_{m+1}b_m b_{m-1} \cdots b_1, \quad b_m \rightsquigarrow c_m b_{m+1} b_m \cdots b_1, \quad m \geq 1, \\ & c_m \rightsquigarrow c_{m+1}d_m d_{m-1} \cdots d_1 e, \quad d_m \rightsquigarrow d_{m+1}d_m \cdots d_1 e, \quad m \geq 1, \\ & e \rightsquigarrow d_1 e, \end{aligned}$$

where $a_m = 0^m$, $b_m = 0^m 1$, $c_m = 0^m 11$, $d_m = 0^m 112$ and $e = 0113$. The number of nodes at level n in $\mathcal{T}[\{110, 021\}]$ equals the number of inversion sequences in $I_n(110, 021)$. From the above generating tree rules, we have that

Corollary 4.2. For all $n \geq 0$, $|I_n(100, 021)| = |I_n(110, 021)|$.

As in the previous subsection, after translating the rules into a system of functional equations and then solving for $\sum_{n \geq 0} |I_n(100, 021)|x^{n+1}$, we obtain the following result, the details are available in an earlier version of the present paper (Mansour and Yıldırım, 2022).

Theorem 4.3. The generating function $R(x) = \sum_{n \geq 0} |I_n(100, 021)|x^{n+1}$ is given by

$$\frac{(1 - 3x)^2}{2x^2\sqrt{1 - 4x}} - \frac{(1 - 3x)(1 - x)}{2x^2}.$$

Moreover, $|I_n(100, 021)| = \frac{n^2+n+6}{2(n+3)(n+2)} \binom{2n+2}{n+1}$ for all $n \geq 0$.

4.3. Class $I_n(102, 021)$

Let $B = \{102, 021\}$. We will apply our algorithm to $I_n(B)$ and characterize the generating tree $\mathcal{T}(B)$ that leads to an exact enumerating formula for this class. We have that the generating tree $\mathcal{T}[B]$ is given by

$$\begin{aligned} \text{Root: } a_0, \text{ Rules: } a_m &\rightsquigarrow a_{m+1}b_mb_{m-1} \cdots b_1, \quad b_m \rightsquigarrow c_mb_{m+1}d_m \cdots d_2e, \quad m \geq 1, \\ c_m &\rightsquigarrow c_{m+1}c_{m+1}, \quad d_m \rightsquigarrow fd_{m+1}d_m \cdots d_1, \\ e &\rightsquigarrow fd_2e, \quad f \rightsquigarrow f, \end{aligned}$$

where $a_m = 0^m$, $b_m = 0^m 1$, $c_m = 0^m 10$, $d_m = 0^m 12$, $e = 0013$, and $f = 00130$.

By similar techniques used in the previous cases, we obtain the following result. For more details on the proof, see Mansour and Yıldırım (2022).

Theorem 4.4. The generating function $R(x) = \sum_{n \geq 0} |I_n(102, 021)|x^{n+1}$ is given by

$$\frac{1 - \sqrt{1 - 4x}}{2x(1 - x)} - \frac{(2x^2 - 2x + 1)(x^3 - 2x^2 + 3x - 1)}{(1 - x)^4(1 - 2x)}.$$

Moreover, $|I_n(102, 021)| = \sum_{k=0}^n \frac{1}{k+1} \binom{2k}{k} - 1 - \frac{1}{6}n^3 - \frac{11}{6}n + 2^n$.

4.4. Class $I_n(100, 012)$

Let $B = \{100, 012\}$. We start with the following lemmas.

Lemma 4.5. Let $m \geq 1$. The generating function $B_m^{(1)}(x)$ for the number of words π' with $n - 1$ letters over alphabet $\{0, 1, \dots, m - 1\}$, $n \geq 1$, such that $0^m m 0 \pi' \in I_{n+m+1}(B)$ is given by $x(1 + x)^{m-1}$.

Proof. Clearly, any inversion sequence $0^m m 0 \pi' \in I_{n+m+1}(B)$ can be decomposed as $0^m m 0 j \pi^{(j)}$ with $j = 1, 2, \dots, m - 1$. Note that the number of inversion sequences $0^m m 0 j \pi^{(j)} \in I_{n+m+1}(B)$ equals the number of inversion sequences $0^j j 0 \theta^{(j)} \in I_{n+j+1}(B)$, where $\theta^{(j)}$ is a word of length $n - 2$ over alphabet $\{0, 1, \dots, j - 1\}$. Hence, $B_m^{(1)}(x) = x + x \sum_{j=1}^{m-1} B_j^{(1)}(x)$, for all $m \geq 1$. By induction on m , we complete the proof. \square

Lemma 4.6. Let $m \geq 1$. The generating function $B_m^{(2)}(x)$ for the number of words π' with $n - 1$ letters over alphabet $\{0, 1, \dots, m\}$, $n \geq 1$, such that $0^m m 0 \pi' \in I_{n+m+1}(B)$ is given by $\frac{x(1+x)^{m-1}}{1-x}$.

Proof. Similar to the proof of Lemma 4.5, we see $B_m^{(2)}(x) = x + xB_m^{(2)}(x) + x \sum_{j=1}^{m-1} B_j^{(1)}(x)$, for all $m \geq 1$. Then, by Lemma 4.5, we complete the proof. \square

Lemma 4.7. Let $m \geq 1$. The generating function $B_m^{(3)}(x)$ for the number of words π' with $n - 1$ letters over alphabet $\{0, 1, \dots, m - 1\}$, $n \geq 1$, such that $0^m m \pi' \in I_{n+m}(B)$ is given by

$$(m + 1)x^3(1 + x)^{m-2} - x(x^2 - 2x - 1)(1 + x)^{m-2}.$$

Proof. Similar to the proof of Lemma 4.5, we see $B_m^{(3)}(x) = x + xB_m^{(1)}(x) + x \sum_{j=1}^{m-1} B_j^{(3)}(x)$, for all $m \geq 1$. Define $B^{(3)}(x, u) = \sum_{m \geq 1} B_m^{(3)}(x)u^m$. By multiplying the above recurrence by u^m and summing over $m \geq 1$ with using Lemma 4.6, we obtain

$$B^{(3)}(x, u) = \frac{xu(1 + x - u - 2ux)}{(1 - u - ux)^2}.$$

Then, by finding the coefficient of u^m , we complete the proof. \square

Lemma 4.8. Let $m \geq 1$. The generating function $B_m(x)$ for the number of words π' with $n - 1$ letters over alphabet $\{0, 1, \dots, m\}$, $n \geq 1$, such that $0^m m \pi' \in I_{n+m}(B)$ is given by

$$\frac{x((m - 1)x^2(1 - x) + x + 1)(1 + x)^{m-2}}{(1 - x)^2}.$$

Proof. Similar to the proof of Lemma 4.5, we see $B_m(x) = x + xB_m^{(2)}(x) + xB_m(x) + x \sum_{j=1}^{m-1} B_j^{(3)}(x)$, for all $m \geq 1$. By Lemmas 4.6 and 4.7, we complete the proof. \square

When we apply our algorithm, we obtain the following generating tree.

Proposition 4.9. Let $\mathcal{T}_m(B)$ be the generating tree for all inversion sequence $\pi = 0^m m \pi'$ that avoids $\{100, 012\}$. The generating tree $\mathcal{T}[B]$ is given by

$$\text{Root: } a_1, \text{ Rules: } a_m \rightsquigarrow a_{m+1}, \mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_m,$$

where $a_m = 0^m$ with $m \geq 1$.

Now, we are ready to find an explicit formula for the generating function

$$R(x) = \sum_{n \geq 0} |I_n(100, 012)|x^{n+1}.$$

Theorem 4.10. The generating function $R(x)$ is given by

$$R(x) = \frac{x(x^6 - x^5 - 3x^4 + x^3 + 3x^2 - 3x + 1)}{(1 - x)^3(1 - x - x^2)^2} \\ = x + 2x^2 + 5x^3 + 12x^4 + 27x^5 + 56x^6 + 110x^7 + 207x^8 + 378x^9 + 675x^{10} + \dots$$

Moreover, by the sequence A001629 in Sloane (1996), for all $n \geq 0$,

$$|I_n(100, 012)| = \frac{(n + 7)Fib_n + 15Fin_{n+1} + nFib_{n+2}}{5} - 1 - \binom{n + 2}{2},$$

where Fib_n is the n th Fibonacci number, see sequence A000045 in Sloane (1996).

Proof. Let $R_m(x)$ be the generating function for the number of nodes in the subtree $\mathcal{T}(B, 0^m)$ of Proposition 4.9. Hence, Proposition 4.9 and Lemma 4.8 give

$$R_m(x) = x + xR_{m+1}(x) + x \sum_{j=1}^m \frac{x((j-1)x^2(1-x) + x + 1)(1+x)^{j-2}}{(1-x)^2}.$$

Define $R(x, u) = \sum_{m \geq 1} R_m(x)u^{m-1}$. Then

$$R(x, u) = \frac{x}{1-u} + \frac{x}{u}(R(x, u) - R(x, 0)) - \frac{(ux^3 - ux^2 + ux + u - 1)x^2}{(1-u)(ux + u - 1)^2(1-x)^2}.$$

Then by applying the kernel method with taking $u = x$, we obtain

$$R(x, 0) = \frac{x(x^6 - x^5 - 3x^4 + x^3 + 3x^2 - 3x + 1)}{(1-x)^3(1-x-x^2)^2},$$

which completes the proof. \square

4.5. Class $I_n(011, 201)$

Let $B = \{011, 201\}$. By applying our algorithm to $I_n(B)$, we obtain the generating tree $\mathcal{T}[B]$ as follows:

$$\begin{aligned} \text{Root: } a_1, \quad \text{Rules: } a_m &\rightsquigarrow a_{m+1}a_m b_{m,2} \cdots b_{m,m}, \\ &b_{m,j} \rightsquigarrow (a_{m+2-j})^2 b_{m+3-j,2} \cdots b_{m,j-1} b_{m,j} \cdots b_{m,m}, \end{aligned}$$

where $a_m = 0^m$ with $m \geq 1$ and $b_{m,j} = 0^m j$ with $2 \leq j \leq m$.

We define $A_m(x)$ and $B_{m,j}(x)$ as the generating functions for the number of nodes in the subtrees $\mathcal{T}(B; a_m)$ and $\mathcal{T}(B; b_{m,j})$, respectively. Thus, the generating tree $\mathcal{T}[B]$, leads to

$$\begin{aligned} A_m(x) &= x + xA_{m+1}(x) + xA_m(x) + x(B_{m,2}(x) + \cdots + B_{m,m}(x)), \quad m \geq 1, \\ B_{m,j}(x) &= x + 2xA_{m+2-j}(x) + x \sum_{i=2}^{j-1} B_{m+1-j+i,i}(x) + x \sum_{i=j}^m B_{m,i}(x), \quad 2 \leq j \leq m. \end{aligned}$$

Define $A(x, v) = \sum_{m \geq 1} A_m(x)v^{m-1}$ and $B(x, v, u) = \sum_{m \geq 2} \sum_{j=2}^m B_{m,j}(x)v^{m-2}u^{m-j}$. Then the above recurrence can be written as

$$A(x, v) = \frac{x}{1-v} + xA(x, v) + \frac{x}{v}(A(x, v) - A(x, 0)) + xvB(x, v, 1), \tag{4.15}$$

$$\begin{aligned} B(x, v, u) &= \frac{x}{(1-v)(1-vu)} + \frac{2x}{uv(1-v)}(A(x, uv) - A(x, 0)) \\ &+ \frac{x}{uv(1-v)}(B(x, v, u) - B(x, v, 0)) + \frac{x}{1-u}(B(x, v, u) - uB(x, uv, 1)). \end{aligned} \tag{4.16}$$

Then by taking $v = \frac{x}{1-x}$ into (4.15), we obtain

Theorem 4.11. *The generating function $\sum_{n \geq 0} |I_n(011, 201)|x^{n+1}$ is equal to $A(x, 0)$ that satisfies the following functional equation*

$$A(x, 0) = \frac{x}{1-2x} + \frac{x^2}{(1-x)^2}B(x, x/(1-x), 1).$$

Applying this theorem, we obtain the first 20 terms of $A(x, 0)$ as 1, 2, 5, 15, 52, 202, 859, 3930, 19095, 97566, 520257, 2877834, 16434105, 96505490, 580864901, 3573876308, 22426075431, 143242527870, 929759705415, 6123822269373.

Here, we conjecture that $|I_n(011, 201)|$ equals the number of set partitions of $\{1, 2, \dots, n\}$ that avoid 3-crossings, see Sequence A108304 in Sloane (1996).

4.6. Class $I_n(120, 210)$

Let $B = \{120, 210\}$. By applying our algorithm to $I_n(B)$, we obtain the generating tree $\mathcal{T}[B]$ as follows:

Root: a_1 ,

Rules: $a_m \rightsquigarrow a_{m+1}b_{m,1} \cdots b_{m,m}$, $b_{m,j} \rightsquigarrow b_{m+1,j} \cdots b_{m+2-j,1}b_{m+1,j}b_{m+1-j,1} \cdots b_{m+1-j,m+1-j}$,

where $a_m = 0^m$ with $m \geq 1$ and $b_{m,j} = 0^m j$ with $1 \leq j \leq m$. It is not hard to prove that this is indeed the generating tree $\mathcal{T}[B]$ by using Lemma 2.1.

We define $A_m(x)$ and $B_{m,j}(x)$ as the generating functions for the number of nodes in the subtrees $\mathcal{T}(B; a_m)$ and $\mathcal{T}(B; b_{m,j})$, respectively. Thus, the generating tree $\mathcal{T}[B]$, leads to

$$A_m(x) = x + xA_{m+1}(x) + x(B_{m,1}(x) + \cdots + B_{m,m}(x)), \quad m \geq 1,$$

$$B_{m,j}(x) = x + x \sum_{i=1}^j B_{m+1-j+i,i}(x) + xB_{m+1,j}(x) + x \sum_{i=1}^{m+1-j} B_{m+1-j,i}(x), \quad 1 \leq j \leq m.$$

We define $A(x, v) = \sum_{m \geq 1} A_m(x)v^{m-1}$ and $B(x, v, u) = \sum_{m \geq 1} \sum_{j=1}^m B_{m,j}(x)v^{m-1}u^{m-j}$. Then the above recurrence can be written as

$$A(x, v) = \frac{x}{1-v} + \frac{x}{v}(A(x, v) - A(x, 0)) + xB(x, v, 1), \tag{4.17}$$

$$B(x, v, u) = \frac{x}{(1-v)(1-vu)} + \frac{x}{uv(1-v)}(B(x, v, u) - B(x, v, 0)) + \frac{x}{uv}(B(x, v, u) - B(x, v, 0)) + \frac{x}{1-v}B(x, uv, 1). \tag{4.18}$$

Then by taking $v = x$ into (4.17), we obtain

Theorem 4.12. *The generating function $\sum_{n \geq 0} |I_n(120, 210)|x^{n+1}$ is equal to $A(x, 0)$ that satisfies the following functional equation*

$$A(x, 0) = \frac{x}{1-x} + xB(x, x, 1).$$

Applying this theorem, we obtain the first 20 terms of $A(x, 0)$ as 1, 2, 6, 23, 102, 499, 2625, 14601, 84847, 510614, 3161964, 20050770, 129718404, 853689031, 5701759424, 38574689104, 263936457042, 1824032887177, 12718193293888, 89386742081688.

5. Restricted growth sequences

In the previous sections, we showed that our algorithmic approach based on generating trees can solve many enumerative questions for inversion sequences with pattern restrictions. This approach can be modified to include enumerative results for restricted growth sequences. A sequence of positive integers $\pi = \pi_1\pi_2 \cdots \pi_n$ is called a *restricted growth sequence* of length n if $\pi_1 = 1$ and $\pi_{j+1} \leq 1 + \max\{\pi_1, \dots, \pi_j\}$ for all $1 \leq j < n$. There is a bijection between these sequences and canonical set partitions. A *set partition* of a set A is a collection of non-empty disjoint subsets, called *blocks*, whose union is the set A . A *k-set partition* is a set partition Π with k blocks and it is denoted by

$\Pi = A_1|A_2|\dots|A_k$. A k -set partition $A_1|A_2|\dots|A_k$ is said to be in *standard form* if the blocks A_i are labelled in such a way that $\min A_1 < \min A_2 < \dots < \min A_k$. The set partition $\Pi = A_1|A_2|\dots|A_k$ can be represented equivalently by the *canonical sequential form* (see Mansour, 2013) $\pi_1\pi_2\dots\pi_n$, where $\pi_i \in [n] = \{1, 2, \dots, n\}$ and $i \in A_{\pi_i}$ for all i . It is easy to verify that a word $\pi \in [k]^n$ is a canonical representation of a k -set partition of $[n]$ in standard form if and only if it is a restricted growth sequence, see (Mansour, 2013) and references therein. Henceforth we identify set partitions with their canonical representations. We denote by \mathcal{P}_n the set of all restricted growth sequences of length n , and denote by $\mathcal{P}_{n,k}$ the set of all restricted growth sequences of length n with maximal letter k . Similarly to $I_n(B)$, we denote $\mathcal{P}_n(B)$ be the set of all restricted growth sequences that avoid all the patterns in B .

For a given set of patterns B , we will construct a pattern-avoidance tree $\mathcal{T}(B)$ for the class of pattern-avoiding restricted growth sequences $\mathcal{P}_B = \cup_{n \geq 0} \mathcal{P}_n(B)$. In the case of restricted growth sequences we define the root to be 1 and the children of $\pi_1\pi_2\dots\pi_{n-1} \in \mathcal{P}_n(B)$ are obtained from the set $\{\pi_1\pi_2\dots\pi_{n-1}j \mid 1 \leq j \leq \max\{\pi_1\pi_2\dots\pi_{n-1}\} + 1\}$.

As in Section 2, let $\mathcal{T}(B; \pi)$ denote the subtree consisting of the restricted growth sequence π as the root and its descendants in $\mathcal{T}(B)$. Then we define an equivalence relation on nodes π, π' of $\mathcal{T}(B)$ whenever $\mathcal{T}(B; \pi) \cong \mathcal{T}(B; \pi')$ in the sense of plane trees. Thus, by taking generating trees $\mathcal{T}(B)$ for restricted growth sequences, our algorithm as described in Section 2 can be reduced to an algorithm for finding $\mathcal{T}[B]$, which is the same tree $\mathcal{T}(B)$ where we replace each node by its equivalence class label. For instance, if $B = \{1212\}$, our algorithm with $D = 4$ leads us to guess that the generating tree $\mathcal{T}\{1212\}$ is given by

$$\text{Root: } 1, \text{ Rules: } 12 \dots m \rightsquigarrow 1, 12, \dots, 12 \dots (m + 1).$$

Note that by similar techniques as in the proof of Lemma 2.1, we see that Lemma 2.1 holds for the case of restricted growth sequences as well.

As in inversion sequences, in the next subsections, we present some applications to our algorithm for restricted growth sequences.

5.1. Pattern 1122

We say that two restricted growth sequences π and π' are equivalent if $|\mathcal{P}_n(\pi)| = |\mathcal{P}_n(\pi')|$ for each n . In Jelínek and Mansour (2008), the authors showed that there are 5 different classes of avoiding a pattern of length four, that is, 1231, 1212, 1122, 1112, and 1111. They also suggested formulas for all the classes except the class 1122. Our algorithm suggests the following result for the class 1122.

Lemma 5.1. *Let $a_k = 12 \dots k$ and $b_{k,j} = 12 \dots kj$ for $j = 1, 2, \dots, k$. Then, the generating tree $\mathcal{T}\{1122\}$ is given by*

$$\text{Root: } a_1, \text{ Rules: } a_k \rightsquigarrow b_{k,1}b_{k,2} \dots b_{k,k}a_{k+1}, \quad b_{k,j} \rightsquigarrow b_{k,1} \dots b_{k,j}b_{k-1,j}^{k-j}b_{k,j}.$$

Proof. We label the root by the restricted growth function $a_1 = 1 \in \mathcal{P}_1$. Clearly, the children of a_k are $b_{k,j}$ with $j = 1, 2, \dots, k$ and a_{k+1} . Moreover, the children of $b_{k,j}$ are $b_{k,j}i \sim b_{k,i}$ with $i = 1, 2, \dots, j$, $b_{k,j}i \sim b_{k-1,j}$ (by removing the letters i) with $i = j + 1, \dots, k$, and $b_{k,j}(k + 1) \sim b_{k,j}$ (by removing the letters $k + 1$). This completes the proof. \square

We can translate the tree rules to a system of equations for the corresponding generating functions. We can obtain the first terms of the sequence $|\mathcal{P}_n(\{1122\})|$ as 1, 1, 2, 5, 14, 42, 133, 441, 1523, 5456, 20209, 77186, and 303296.

5.2. Pattern $\{12313, 12323\}$

Based on the algorithm's output, we obtain the succession rules of the generating tree $\mathcal{T}\{12313, 12323\}$. We omit the details of the proof, since it is similar to the proof of Lemma 5.1.

Lemma 5.2. Let $a_k = 12 \cdots k$. Then, the generating tree $\mathcal{T}[\{12313, 12323\}]$ is given by

$$\text{Root : } a_1, \text{ Rules : } a_1 \rightsquigarrow a_1 a_2, \quad a_k \rightsquigarrow a_2^2 a_3 \cdots a_{k+1}, \quad k \geq 2.$$

Let's define $A_k(x)$ to be the generating function for the number of nodes in the subtree $\mathcal{T}(\{12313, 12323\}; a_k)$, we have $A_1(x) = x + xA_1(x) + xA_2(x)$ and $A_k(x) = x + 2xA_2(x) + xA_3(x) + \cdots + xA_{k+1}(x)$. Hence,

$$A(x, v) = \frac{x}{1-v} + \frac{x}{1-v} A_2(x) + \frac{x}{1-v} A(x, v) + \frac{x}{v} (A(x, v) - A(x, 0)),$$

where $A(x, v) = \sum_{k \geq 2} A_k(x) v^{k-2}$. By taking $v = \frac{1-\sqrt{1-4x}}{2}$, we obtain that

$$A_2(x) = A(x, 0) = \frac{4x - 1 + \sqrt{1 - 4x}}{2(1 - 4x)},$$

which leads to the generating function for the number of restricted growth functions of length n that avoid $\{12313, 12323\}$ which is equal to

$$1 + A_1(x) = \frac{1}{1-x} + \frac{x}{2(1-x)} \left(\frac{1}{\sqrt{1-4x}} - 1 \right).$$

Moreover, we also have the following result.

Lemma 5.3. Let $a_k = 12 \cdots k$. Then, the generating tree $\mathcal{T}[\{12313, 12323, 12333\}]$ is given by

$$\text{Root : } a_1, \text{ Rules : } a_1 \rightsquigarrow a_1 a_2, \quad a_2 \rightsquigarrow a_2^2 a_3, \quad a_k \rightsquigarrow a_2^3 a_3 \cdots a_{k-1} a_{k+1}, \quad k \geq 3.$$

As before, Lemma 5.3 leads to the assertion that the generating function for the number of restricted growth functions of length n that avoid $\{12313, 12323, 12333\}$ is given by

$$1 + \frac{x(3 - 9x + \sqrt{1 - 2x - 3x^2})}{2(2 - 7x)(1 - x)}.$$

5.3. Pattern $12 \cdots \ell 1$

By applying our algorithm for the case $T[\{12 \cdots \ell 1\}]$, we see that the generating tree $T[\{12 \cdots \ell 1\}]$ is given by

$$\text{Root : } 1, \text{ Rules : } a_k \rightsquigarrow (a_k)^k, a_{k+1}, \text{ for } k = 1, 2, \dots, \ell - 2, \\ a_{\ell-1} \rightsquigarrow (a_{\ell-1})^\ell,$$

where $a_k = 12 \cdots k$. We can easily verify this. We label the root by the restricted growth function $1 \in P_1$. Clearly, the children of a_k are $a_k j \sim a_k$ for $j = 1, 2, \dots, k$ and a_{k+1} , where $k \leq \ell - 2$. Thus it remains to find the children of $a_{\ell-1}$, which are $a_{\ell-1} j \sim a_{\ell-1}$ with $j = 1, 2, \dots, \ell - 1$ and $a_{\ell-1} \ell \sim a_{\ell-1}$ (by removing the letter 1 because we avoid $12 \cdots \ell 1$). This completes the proof.

To find a formula for the generating function $\sum_{n \geq 1} |\mathcal{P}_n(\{12 \cdots \ell 1\})| x^n$, we define $A_m(x)$ to be the generating function for the number of nodes in the subtrees $\mathcal{T}(\{12 \cdots \ell\}; a_m)$. Hence, by the rules of $\mathcal{T}[\{12 \cdots \ell 1\}]$, we have that $A_k(x) = x + kxA_k(x) + xA_{k+1}(x)$ with $k = 1, 2, \dots, \ell - 2$, and $A_{\ell-1}(x) = x + \ell x A_{\ell-1}(x)$. By induction on k , we have

$$A_k(x) = \frac{x^{\ell-k} (1 - (\ell - 1)x)}{\prod_{j=k}^{\ell} (1 - jx)} + \sum_{i=1}^{\ell-1-k} \frac{x^i}{\prod_{j=k}^{k-1+i} (1 - jx)},$$

which implies the following result.

Theorem 5.4. The generating function $\sum_{n \geq 1} |\mathcal{P}_n(\{12 \cdots \ell 1\})| x^n$ is given by

$$\frac{x^{\ell-1}(1 - (\ell - 1)x)}{\prod_{j=1}^{\ell} (1 - jx)} + \sum_{i=1}^{\ell-2} \frac{x^i}{\prod_{j=1}^i (1 - jx)}.$$

Declaration of competing interest

There are no potential conflicts of interest that the authors have to report.

Data availability

No data was used for the research described in the article.

Acknowledgement

We thank one anonymous referee whose suggestions improved the presentation of the paper.

References

- Auli, J.S., Elizalde, S., 2021. Wilf equivalences between vincular patterns in inversion sequences. *Appl. Math. Comput.* 388, 125514.
- Banderier, C., Bousquet-Mélou, M., Denise, A., Flajolet, P., Gardy, D., Gouyou-Beauchamps, D., 2002. Generating functions for generating trees. *Discrete Math.* 246 (1–3), 29–55.
- Beaton, N.R., Bouvel, M., Guerrini, V., Rinaldi, S., 2019. Enumerating five families of pattern-avoiding inversion sequences; and introducing the powered Catalan numbers. *Theor. Comput. Sci.* 777, 69–92.
- Bouvel, M., Guerrini, V., Rechnitzer, A., Rinaldi, S., 2018. Semi-Baxter and strong-Baxter: two relatives of the Baxter sequence. *SIAM J. Discrete Math.* 32 (4), 2795–2819.
- Brändén, P., Mansour, T., 2005. Finite automata and pattern avoidance in words. *J. Comb. Theory, Ser. A* 110, 127–145.
- Cao, W., Jin, E.Y., Lin, Z., 2019. Enumeration of inversion sequences avoiding triples of relations. *Discrete Appl. Math.* 260, 86–97.
- Chern, S., 2023. On 0012-avoiding inversion sequences and a conjecture of Lin and Ma. *Quaest. Math.* 46 (3), 681–694.
- Corteel, S., Martínez, M.A., Savage, C.D., Weselcouch, M., 2016. Patterns in inversion sequences I. *Discrete Math. Theor. Comput. Sci.* 18 (2).
- Duncan, P., Steingrímsson, E., 2011. Pattern avoidance in ascent sequences. *Electron. J. Comb.* 18 (1), P226.
- Hong, L., Li, R., 2022. Length four pattern avoidance in inversion sequences. *Electron. J. Comb.* 29 (4), P4.37.
- Jelínek, V., Mansour, T., 2008. On pattern-avoiding partitions. *Electron. J. Comb.* 15, R39.
- Kitaev, S., 2011. *Patterns in Permutations and Words*. Monographs in Theoretical Computer Science. Springer.
- Lin, Z., 2018. Restricted inversion sequences and enhanced 3-noncrossing partitions. *Eur. J. Comb.* 70, 202–211.
- Lin, Z., 2020. Patterns of relation triples in inversion and ascent sequences. *Theor. Comput. Sci.* 804, 115–125.
- Lin, Z., Fu, S., 2021. On 120-avoiding inversion and ascent sequences. *Eur. J. Comb.* 93, 103282.
- Lin, Z., Yan, S.H.F., 2020. Vincular patterns in inversion sequences. *Appl. Math. Comput.* 364, 124672.
- Mansour, T., 2013. *Combinatorics of Set Partitions*. CRC Press.
- Mansour, T., 2023. Generating trees for 0021-avoiding inversion sequences and a conjecture of Hong and Li. *Discrete Math. Lett.* 12, 11–14.
- Mansour, T., Shattuck, M., 2015. Pattern avoidance in inversion sequences. *Pure Math. Appl.* 25 (2), 157–176.
- Mansour, T., Shattuck, M., 2022. Further enumeration results concerning a recent equivalence of restricted inversion sequences. *Discrete Math. Theor. Comput. Sci.* 24 (1), 4.
- Mansour, T., Yıldırım, G., 2022. An algorithmic approach based on generating trees for enumerating pattern-avoiding inversion sequences. [arXiv:2210.04489](https://arxiv.org/abs/2210.04489).
- Martínez, M.A., Savage, C.D., 2018. Patterns in inversion sequences II: inversion sequences avoiding triples of relations. *J. Integer Seq.* 21, 18.2.2.
- Nakamura, B., Zeilberger, D., 2013. Using Noonan-Zeilberger functional equations to enumerate (in polynomial time!) generalized Wilf classes. *Adv. Appl. Math.* 50 (3), 356–366.
- Sloane, N.J.A., 1996. *The On-Line Encyclopedia of Integer Sequences*. <http://oeis.org>.
- Testart, B., 2022. Inversion sequences avoiding the pattern 010. [arXiv:2212.07222v2](https://arxiv.org/abs/2212.07222v2).
- Vatter, V., 2008. Enumeration schemes for restricted permutations. *Comb. Probab. Comput.* 17 (1), 137–159.
- West, J., 1996. Generating trees and forbidden subsequences. *Discrete Math.* 157, 363–374.
- Yan, C., Lin, Z., 2020. Inversion sequences avoiding pairs of patterns. *Discrete Math. Theor. Comput. Sci.* 22 (1), 23.
- Yang, M., Zeilberger, D., 2020. Increasing consecutive patterns in words. *J. Algebraic Comb.* 51 (1), 89–101.
- Zeilberger, D., 1998. Enumeration schemes and, more importantly, their automatic generation. *Ann. Comb.* 2 (2), 185–195.