Enumeration schemes for vincular patterns

Andrew M. Baxter [a,1], Lara K. Pudwell [b,*]

[a] Mathematics Department, Pennsylvania State University, State College, PA 08902, United States

[b] Department of Mathematics and Computer Science, Valparaiso University, Valparaiso, IN 46383, United States

ARTICLE INFO

ABSTRACT

We extend the notion of an enumeration scheme developed by Zeilberger and Vatter to the case of vincular patterns (also called "generalized patterns" or "dashed patterns"). In particular we provide an algorithm which takes in as input a set B of vincular patterns and search parameters and returns a recurrence (called a "scheme") to compute the number of permutations of length n avoiding B or confirmation that no such scheme exists within the search parameters. We also prove that if B contains only consecutive patterns and patterns of the form $\sigma_1 \sigma_2 \ldots \sigma_{t-1} - \sigma_t$, then such a scheme must exist and provide the relevant search parameters. The algorithms are implemented in Maple and we provide empirical data on the number of small pattern sets admitting schemes. We make several conjectures on Wilf-classification based on this data. We also outline how to refine schemes to compute the number of B-avoiding permutations of length n with k inversions.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Enumeration schemes are special recurrences which were originally designed to compute the number of permutations avoiding a set of classical patterns. In the current work we extend the tools of enumeration schemes to compute the number of permutations avoiding a set of *vincular* patterns.

Let $[n] = \{1, 2, \ldots, n\}$. For a word $w \in [n]^k$, we write $w = w_1 w_2 \ldots w_k$ and define the *reduction* red(w) to be the word obtained by replacing the ith smallest letter(s) of w with i. For example red$(839183) = 324132$. If red$(u) = $ red(w), we say that u and w are *order-isomorphic* and write $u \sim w$.

Let $\mathcal{S}_n$ be the set of permutations of length n. We say that permutation $\pi \in \mathcal{S}_k$ contains $\sigma \in \mathcal{S}_k$ as a *classical pattern* if there is some k-tuple $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ such that red$(\pi_{i_1} \pi_{i_2} \ldots \pi_{i_k}) = \sigma$. The subsequence $\pi_{i_1} \pi_{i_2} \ldots \pi_{i_k}$ is called a *copy* (or *occurrence*) of σ. If π does not contain σ, then π is said to *avoid* σ. Hence we see that $\pi = 34512$ contains 231 as a classical pattern witnessed by the subsequence $\pi_1 \pi_3 \pi_4 = 351$, but exhaustive checking shows π avoids 132. The subset of $\mathcal{S}_n$ consisting of permutations avoiding σ is denoted $\mathcal{S}_n(\sigma)$. For a set of patterns B, π is said to avoid B if π avoids all $\sigma \in B$, and we denote the set of B-avoiding permutations by

$$\mathcal{S}_n(B) := \bigcap_{\sigma \in B} \mathcal{S}_n(\sigma). \tag{1}$$

We will denote the size of $\mathcal{S}_n(B)$ by $s_n(B) = |\mathcal{S}_n(B)|$.

Vincular patterns resemble classical patterns, with the constraint that some of the letters in a copy must be consecutive. Formally, a *vincular pattern of length k* is a pair (σ, X) where σ is a permutation in $\mathcal{S}_k$ and $X \subseteq \{0, 1, 2, \ldots, k\}$ is a set of

* Corresponding author. Tel.: +1 219 464 5414; fax: +1 219 464 5361.
  E-mail addresses: baxter@math.psu.edu (A.M. Baxter), Lara.Pudwell@valpo.edu (L.K. Pudwell).
[1] Tel.: +1 732 801 9133; fax: +1 814 865 3735.

"adjacencies". A permutation $\pi \in \mathcal{S}_n$ contains the vincular pattern (σ, X) if there is a k-tuple $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ such that the following three criteria are satisfied:

- $\operatorname{red}(\pi_{i_1} \pi_{i_2} \ldots \pi_{i_k}) = \sigma$.
- $i_{x+1} = i_x + 1$ for each $x \in X \setminus \{0, k\}$.
- $i_1 = 1$ if $0 \in X$ and $i_k = n$ if $k \in X$.

In the present work we restrict our attention to patterns (σ, X) where $\sigma \in \mathcal{S}_k$ and $X \subseteq [k-1]$, rendering the third containment criterion moot.[2] The subsequence $\pi_{i_1} \pi_{i_2} \ldots \pi_{i_k}$ is called a *copy* of (σ, X). In the permutation $\pi = 162534$, the subsequence 1253 is a copy of $(1243, \{3\})$, but the subsequence 1254 is not a copy since the 5 and 4 are not adjacent in π. The classical pattern σ is precisely the vincular pattern $(\sigma, \emptyset)$ since no adjacencies are required, while the consecutive pattern σ is the vincular pattern $(\sigma, [k-1])$ since all internal adjacencies are required.

In practice we write (σ, X) as a permutation with a dash between σ_j and σ_{j+1} if $j \notin X$. Thus we will often refer to "the vincular pattern σ" without explicitly referring to X. For example, $(1243, \{3\})$ is written 1-2-43.

If the permutation π does not contain a copy of the vincular pattern σ, then π is said to *avoid* σ. We will use the same notation $\mathcal{S}_n(\sigma)$ to denote the set of permutations avoiding the vincular pattern σ, and similarly $\mathcal{S}_n(B)$ denotes those permutations avoiding every vincular pattern $\sigma \in B$.

Observe that a vincular pattern (σ, X) of length k exhibits similar symmetries to those of permutations. The reverse is given by $(\sigma, X)^r = (\sigma^r, k - X)$ where $k - X = \{k - x : x \in X\}$. The complement is $(\sigma, X)^c = (\sigma^c, X)$. It follows that that π avoids σ if and only if π^r avoids σ^r. Similarly, π avoids σ if and only if π^c avoids σ^c. To consider inverses, one must generalize to the bivincular patterns introduced in [6] which incorporate adjacency restrictions on not only the indices of letters forming a forbidden pattern, but also on the values of the offending letters as well. Since the inverse of a vincular pattern is not itself a vincular pattern, we will disregard inverses.

Vincular patterns were introduced as "generalized patterns" by Babson and Steingrímsson in [2] as a generalization of classical patterns as part of a systematic search for Mahonian permutation statistics. They soon took on a life of their own spawning numerous papers, including [7,9,12,13]. They have also been called "dashed patterns" to distinguish them from other generalizations of classical patterns [22], but Claesson has since dubbed them "vincular patterns" to connect them with the bivincular patterns introduced in [6]. See Steingrímssson's survey for a fuller history in [20]. They have been linked to many of the common combinatorial structures such as the Catalan and Bell numbers as well several rarer or as-yet unseen structures.

The present work focuses on enumeration schemes, which were introduced by Zeilberger in [23] as an automated method to compute $s_n(B)$ for many different B. Vatter improved schemes in [21] with the introduction of gap vectors, and Zeilberger provided an alternative implementation in [24]. The greatest feature of schemes is that they may be discovered automatically by a computer: the user need only input the set B (along with bounds to the computer search) and the computer will return an enumeration scheme (if one exists within the bounds of the search) which computes $s_n(B)$ in polynomial time. The second author extended these methods to consider pattern avoidance in permutations of a multiset in [16,17], as well as barred-pattern avoidance in [18].

Section 2 provides an overview of how enumeration schemes work and constructs a scheme for 23-1-avoiding permutations by hand. Section 3 outlines how the discovery of schemes can be done via a finite computer search. Section 4 demonstrates instances where we are guaranteed a successful search for a scheme. Section 5 is divided into three subsections. The first provides an analysis of the algorithm's success rate in discovering schemes automatically, the second outlines the implications for Wilf-classification of vincular patterns, and the third gives an example of how enumeration schemes for vincular patterns may be adapted to count according to inversion number as per [3].

2. An overview of enumeration schemes

Broadly, enumeration schemes are succinct encodings for a system of recurrence relations to compute the cardinalities for a family of sets. The enumerated sets are subsets of $\mathcal{S}_n(B)$ determined by prefixes. For a pattern $p \in \mathcal{S}_k$, let $\mathcal{S}_n(B)[p]$ be the set of permutations $\pi \in \mathcal{S}_n(B)$ such that $\operatorname{red}(\pi_1 \pi_2 \ldots \pi_k) = p$. For further refinement, let $w \in \{1, 2, \ldots, n\}^k$ and define the set

$$\mathcal{S}_n(B)[p; w] = \{\pi \in \mathcal{S}_n(B)[p] : \pi_i = w_i \text{ for } 1 \leq i \leq k\}.$$

For an example, consider

$$\mathcal{S}_5(1\text{-}2\text{-}3)[21; 53] = \{53142, 53214, 53241, 53412, 53421\}.$$

Clearly this refinement is worthwhile only when $\operatorname{red}(w) = p$. The redundancy of including p in the $\mathcal{S}_n(B)[p; w]$ notation is maintained to emphasize the subset relation. We will denote sizes of these sets by $s_n(B)[p] = |\mathcal{S}_n(B)[p]|$ and $s_n(B)[p; w] = |\mathcal{S}_n(B)[p; w]|$.

---

[2] We enact this restriction partly for simplicity. It is plausible that the prefix-focused arguments below extend to patterns (σ, X) with $0 \in X$ with few modifications, but it is unlikely such an approach could work if $k \in X$.

By looking at the prefix of a permutation, one can identify likely "trouble spots" where forbidden patterns may appear. For example, suppose we wish to avoid the pattern 23-1. Then the presence of the pattern 12 in the prefix indicates the potential for the entire permutation to contain a 23-1 pattern. In [21], Vatter partitions $\mathcal{S}_n(B)$ according to the inverse notion of the pattern formed by the *smallest* k letters in $\pi \in \mathcal{S}_n(B)$. This partition is not well-suited for keeping track of adjacencies.

Enumeration schemes take a divide-and-conquer approach to enumeration. For a permutation $p \in \mathcal{S}_k$, we say that $p' \in \mathcal{S}_{k+1}$ is a *child* of p if $p'_1 p'_2 \cdots p'_k \sim p$. For example, the children of $p = 312$ are 3124, 4123, 4132, and 4231. Any set $\mathcal{S}_n(B)[p]$ for $p \in \mathcal{S}_k$ may be partitioned into the family of sets $\mathcal{S}_n(B)[p']$ for each of the children $p' \in \mathcal{S}_{k+1}(B)[p]$. These smaller sets are then counted as described below, and their sizes are totaled to obtain $s_n(B)[p]$. In the end we will have computed $s_n(B)$, since $\mathcal{S}_n(B) = \mathcal{S}_n(B)[\epsilon] = \mathcal{S}_n(B)[1]$ for $n \geq 1$, where ϵ is the empty permutation.

For a prefix pattern $p \in \mathcal{S}_k$, we will classify $\mathcal{S}_n(B)[p]$ in one of three ways:

(1) If $n = k$, then $\mathcal{S}_n(B)[p]$ is either $\{p\}$ or $\emptyset$, depending on whether p avoids B.
(2) For each $w \in [n]^k$ such that red$(w) = p$ one of the following happens:
   (2a) $\mathcal{S}_n(B)[p; w]$ is empty, or
   (2b) $\mathcal{S}_n(B)[p; w]$ is in bijection with some $\mathcal{S}_{n'}(B)[p'; w']$ for $n' < n$.
(3) $\mathcal{S}_n(B)[p]$ must be partitioned further, so $s_n(B)[p] = \sum_{p' \in \mathcal{S}_{k+1}(B)[p]} s_n(B)[p']$.

Case (1) provides the base cases for our recurrence. For case (2), if there is any w for which neither (2a) nor (2b) holds then we must divide $\mathcal{S}_n(B)[p]$ as in case (3). For case (2a), the *gap vector criteria* for the given p identify which w yield empty $\mathcal{S}_n(B)[p; w]$. Gap vector criteria are developed in Section 2.1. The bijection in (2b) is performed by removing a certain subset of the first k letters of $\pi \in \mathcal{S}_n(B)[p; w]$, and which subset may be "nicely" removed depends on p and B but not w. Such subsets are called *reversibly deletable*, and are developed in Section 2.2.

2.1. Gap vectors

The motivation for gap vectors lies in the idea of "vertical space" (in the sense of the graph of a permutation) in a prefix w. Sometimes the difference of the values of letters in the prefix is so great that a forbidden pattern *must* appear. To make this notion more precise, we follow our example above and compute $s_n(B)$ for $B = \{23\text{-}1\}$. Observe that $\mathcal{S}_n(B)[12; w_1 w_2]$ is empty if $w_1 > 1$, since if $\pi \in \mathcal{S}_n(B)[12; w_1 w_2]$ then $\pi_i = 1$ for some $i \geq 3$. Thus red$(w_1 w_2 \pi_i) = 231$ and so π contains 23-1. Hence $\mathcal{S}_n(B)[12; w]$ is non-empty only if $1 = w_1 < w_2 \leq n$.

For a word $w \in [n]^k$, let c_i be the ith smallest letter of w, $c_0 = 0$ and $c_{k+1} = n + 1$. Define the $(k+1)$-vector $\vec{g}(n, w)$ such that the ith component $g_i = c_i - c_{i-1} - 1$. We call $\vec{g}(n, w)$ the *spacing vector* of w. Observe that for any $\pi \in \mathcal{S}_n(B)[p; w]$, g_i is the number of letters in $\pi_{k+1}, \pi_{k+2}, \ldots, \pi_n$ which lie between c_{i-1} and c_i, and so $\vec{g}(n, w)$ indicates what letters follow the prefix. In the preceding paragraph, we saw that $\mathcal{S}_n(B)[12; w]$ is empty if $\vec{g}(n, w) \geq \langle 1, 0, 0 \rangle$ where $\geq$ represents the product order for $\mathbb{N}^3$ (component-wise dominance). Towards generality, we make the following definition.

Definition 1. Given a set of forbidden patterns B and prefix $p \in \mathcal{S}_k$, then $\vec{v} \in \mathbb{N}^{k+1}$ is a *gap vector for prefix p with respect to B* if for all n, $\mathcal{S}_n(B)[p; w] = \emptyset$ for all w such that $\vec{g}(n, w) \geq \vec{v}$. When this happens, we say that w *satisfies* the *gap vector criterion* for $\vec{v}$.

It should be noted that this definition reverses the terminology of [21] to match that of [3,16–18,24].

From this definition we see $\vec{v} = \langle 1, 0, 0 \rangle$ is a gap vector for $p = 12$ with respect to $B = \{23\text{-}1\}$, and any prefix $w = w_1 w_2$ with $1 < w_1 < w_2 \leq n$ satisfies the gap vector condition for $\vec{v}$.

Observe that gap vectors for a given prefix $p \in \mathcal{S}_k$ form an upper order ideal in $\mathbb{N}^{k+1}$, i.e., if $\vec{u} \geq \vec{v}$ for gap vector $\vec{v}$ then $\vec{u}$ is also a gap vector. Hence it suffices to determine only the minimal elements since they will form a basis (since $\mathbb{N}^{k+1}$ is partially well-ordered). Details of the automated discovery of gap vectors are left to Section 3.1.

2.2. Reversible deletions

If we are considering $\mathcal{S}_n(B)[p; w]$ for a w that fails all gap vector criteria, we rely on bijections with previously-computed $\mathcal{S}_{n'}(B)[p'; w']$ for $n' < n$. To continue the example above, consider $\mathcal{S}_n(23\text{-}1)[12; 1w_2]$. An initial $\pi_1 = 1$ cannot take part in a 23-1 pattern, so the map of deleting π_1 is a bijection

$$d_1 : \mathcal{S}_n(23\text{-}1)[12; 1w_2] \to \mathcal{S}_{n-1}(23\text{-}1)[1; w_2 - 1]$$

where $d_1 : \pi_1 \pi_2 \ldots \pi_n \mapsto \text{red}(\pi_2 \pi_3 \ldots \pi_n)$. Hence we see that $s_n(23\text{-}1)[12; 1w_2] = s_{n-1}(23\text{-}1)[1; w_2 - 1]$.

More generally define the deletion $d_r(\pi) := \text{red}(\pi_1 \ldots \pi_{r-1} \pi_{r+1} \ldots \pi_n)$, that is, the permutation obtained by omitting the rth letter of π and reducing. Furthermore for a set R, define $d_R(\pi)$ to be the permutation obtained by deleting π_r for each $r \in R$ and then reducing. For a word w with no repeated letters, define $d_r(w)$ to be the word obtained by deleting the rth letter and then subtracting 1 from each letter larger than w_r. Similarly, to construct $d_R(w)$ delete w_r for each $r \in R$ and subtract $|\{r \in R : w_r < w_i\}|$ from w_i. For example $d_3(6348) = 537$ and $d_{\{1,3\}}(6348) = 36$. It can be seen that this definition is equivalent to the one given above when $w \in \mathcal{S}_k$, and it allows for more succinct notation in the upcoming definition. For any set R and $n \geq |R|$, $d_R : \mathcal{S}_n(\emptyset)[p; w] \to \mathcal{S}_{n-|R|}(\emptyset)[d_R(p); d_R(w)]$ is a bijection. Sometimes we are lucky and the restriction to $\mathcal{S}_n(B)[p; w]$ is a bijection with $\mathcal{S}_{n-|R|}(B)[d_R(p); d_R(w)]$, leading to the following definition.

Definition 2. The index r is *reversibly deletable for p with respect to B* if the map

$$d_r : \mathcal{S}_n(B)[p; w] \to \mathcal{S}_{n-1}(B)[d_r(p); d_r(w)]$$

is a bijection for all w failing the gap vector criterion for every gap vector of p with respect to B (i.e., d_r is a bijection for all w such that $\mathcal{S}_n(B)[p; w] \neq \emptyset$).

The set of indices R is *reversibly deletable for p with respect to B* if the map

$$d_R : \mathcal{S}_n(B)[p; w] \to \mathcal{S}_{n-|R|}(B)[d_R(p); d_R(w)]$$

is a bijection for all w failing the gap vector criterion for every gap vector of p with respect to B (i.e., d_R is a bijection for all w such that $\mathcal{S}_n(B)[p; w] \neq \emptyset$).

Note that the empty set $R = \emptyset$ is always reversibly deletable. We are interested in finding non-empty reversibly deletable sets when they exist. Also observe that if the prefix p contains a forbidden pattern then $\mathcal{S}_n(B)[p; w] = \emptyset$ for any appropriate w, and so $\vec{0} = \langle 0, 0, \ldots, 0 \rangle$ is a gap vector. Furthermore if $\vec{0}$ is a gap vector then any set $R \subseteq \{1, 2, \ldots, |p|\}$ is vacuously reversibly deletable.

In [21] Vatter uses the term ES^+-*reducible* to describe p for which there is a non-empty reversibly deletable set. When there is no non-empty reversibly deletable set, then p is called ES^+-*irreducible*. We will not make use of this terminology in the current work.

In the classical case, Vatter proved that identifying reversibly deletable indices is a finite process in [21]. We will prove the analogous result for vincular patterns in Section 3.2.

2.3. Formal definition of an enumeration scheme

Formally, an enumeration scheme E for $\mathcal{S}_n(B)$ is a set of triples (p, G_p, R_p), where $p \in \mathcal{S}_k$ is a prefix pattern, G_p is a basis of gap vectors for p with respect to B, and R_p is a reversibly deletable set for p with respect to B. Furthermore, E must satisfy the following criteria:

1. $(\epsilon, \emptyset, \emptyset) \in E$.
2. For any $(p, G_p, R_p) \in E$,
   (a) If $R_p = \emptyset$ and $\vec{0} \notin G_p$, then $(p', G_{p'}, R_{p'}) \in E$ for every child p' of p.
   (b) If $R_p \neq \emptyset$, then $(\hat{p}, G_{\hat{p}}, R_{\hat{p}}) \in E$ for $\hat{p} = d_{R_p}(p)$.

One can then "read" the enumeration scheme E to compute $s_n(B)[p; w]$ according to the following rules:

1. If w satisfies the gap vector criteria for some $\vec{v} \in G_p$, then $s_n(B)[p; w] = 0$.
2. For each prefix w that fails the gap criteria for all $\vec{v} \in G_p$, $s_n(B)[p; w] = s_{n-|R_p|}(B)[d_{R_p}(p); d_{R_p}(w)]$ (i.e., R_p is a reversibly deletable set of indices).
3. If $R_p = \emptyset$ then $s_n(B)[p] = \sum_{p' \in \mathcal{S}_{k+1}(B)[p]} s_n(B)[p']$.

When combined with the obvious initial condition that $s_n(B)[p; w] = 1$ when p has length n and avoids B, the scheme provides a system of recurrences to compute $s_n(B)[p; w]$ and hence $s_n(B)$.

As an example consider $\mathcal{S}_n(23\text{-}1)$, discussed above, with the enumeration scheme:

$$E = \{(\epsilon, \emptyset, \emptyset), (1, \emptyset, \emptyset), (12, \{\langle 1, 0, 0 \rangle\}, \{1\}), (21, \emptyset, \{1\})\}. \tag{2}$$

The definition of schemes implies $(\epsilon, \emptyset, \emptyset) \in E$, and $R_\epsilon = \emptyset$ requires $(1, G_1, R_1) \in E$. Starting with the pattern 1 yields no additional information so G_1 and R_1 are both empty. Thus we see $(12, G_{12}, R_{12}) \in E$ and $(21, G_{21}, R_{21}) \in E$. As discussed in Section 2.1, $\langle 1, 0, 0 \rangle \in G_{12}$. It is easily seen this forms a basis for all gap vectors for 12 and so $G_{12} = \{\langle 1, 0, 0 \rangle\}$. As discussed in Section 2.2, $R_{12} = \{1\}$.

Moving on to the prefix pattern $p = 21$, it can be seen that $R_{21} = \{1\}$ by the following argument. Suppose $\pi \in \mathcal{S}_n(\emptyset)[21]$. First observe that deleting π_1 cannot create a 23-1 which was not already present in $\pi_2 \cdots \pi_n$. Next π_1 cannot take part in a 23-1 pattern since this would require $\text{red}(\pi_1 \pi_2) = \text{red}(23) = 12$ while it is known that $\pi_1 > \pi_2$. Hence the map d_1 restricts to a bijection $\mathcal{S}_n(23\text{-}1)[21] \to \mathcal{S}_{n-1}(23\text{-}1)[1]$, so we may let $R_{21} = \{1\}$. Since this argument can hold regardless of the actual letters π_1 and π_2, we may let $G_{21} = \emptyset$. This completes the construction of E above.

The scheme E translates into the following system of recurrences:

$$\begin{aligned}
s_n(23\text{-}1) &= s_n(23\text{-}1)[\epsilon] \\
&= s_n(23\text{-}1)[1] \\
&= \sum_{a=1}^{n} s_n(23\text{-}1)[1; a]
\end{aligned}$$

$$s_n(23\text{-}1)[1; a] = \sum_{b=1}^{a-1} s_n(23\text{-}1)[21; ab] + \sum_{b=a+1}^{n} s_n(23\text{-}1)[12; ab]$$

$$s_n(\text{23-1})[21; ab] = s_{n-1}(\text{23-1})[1; b]$$

$$s_n(\text{23-1})[12; ab] = \begin{cases} s_{n-1}(\text{23-1})[1; b - 1], & a = 1 \\ 0, & a > 1. \end{cases}$$

This system simplifies to:

$$s_n(\text{23-1})[1; a] = \begin{cases} \displaystyle\sum_{b=1}^{n-1} s_{n-1}(\text{23-1})[1; b], & a = 1 \\ \displaystyle\sum_{b=1}^{a-1} s_{n-1}(\text{23-1})[1; b], & 1 < a \leq n \end{cases}$$

which can be used to compute arbitrarily many terms of the sequence $s_n(\text{23-1})$ in polynomial time.

Claesson shows in Proposition 3 of [7] that $s_n(\text{23-1})$ is the nth Bell number, and his bijection also implies that $s_n(\text{23-1})[1; a]$ is the number of partitions of $[n]$ such that a is the largest letter in the same block as 1. The triangle formed is an augmented version of Aitken's array as described in OEIS sequence A095149 [1].

If $|E|$ is finite, we say that B admits a finite enumeration scheme. The length of the longest p appearing is called the *depth* of E. Not every set B admits a finite enumeration scheme, the simplest example being the classical pattern 2-3-1. Let E_{231} be the scheme for $\mathcal{S}_n(\text{2-3-1})$, and let $J_t = t(t-1)\cdots 21$ be the decreasing permutation of length t. It can be shown that $G_{J_t} = \emptyset$ and $R_{J_t} = \emptyset$ for any t, and hence E_{231} contains the triple $(J_t, \emptyset, \emptyset)$ for all $t \geq 1$ and hence is infinite. It should be noted, however, that the enumeration scheme for $\mathcal{S}_n(\text{1-3-2})$ is finite (of depth 2) and $s_n(\text{2-3-1}) = s_n(\text{1-3-2})$ by symmetry.

In general, if B admits an enumeration scheme E_B of depth d then its set of complements $B^c = \{\sigma^c : \sigma \in B\}$ also admits an enumeration scheme E_{B^c} of depth d. In fact, one can say $(p, R, G) \in E_B$ if and only if $(p^c, R, G^r) \in E_{B^c}$ where $G^r = \{\langle g_{k+1}, g_k, \ldots, g_1 \rangle : \langle g_1, g_2, \ldots, g_{k+1} \rangle \in G\}$. This follows directly from the definitions given above and is left to the reader. One cannot make analogous statements regarding $B^r = \{\sigma^r : \sigma \in B\}$, and so B may not have a finite scheme while B^r does.

3. Automated discovery

We now turn to the process of automating the discovery of enumeration schemes for vincular patterns, since this automation is the most outstanding feature of this method. The overall algorithm proceeds as follows.

Algorithm 3. 1. Initialize $E := \{(\epsilon, \emptyset, \emptyset)\}$.
2. Let P be the set of all children of all prefixes p such that $(p, G_p, R_p) \in E$ and $R_p = \emptyset$ and $\vec{0} \notin G_p$. If there are no such prefixes, return E. Otherwise proceed to step 3.
3. For each $p \in P$, find a basis of gap vectors G_p.
4. For each $p \in P$, find a non-empty reversibly deletable set of indices R_p given the gap vector criteria in G_p. If no such R_p exists, let $R_p = \emptyset$.
5. Let $E = E \cup \{(p, G_p, R_p) : p \in P\}$.
6. Return to step 2.

Steps 1, 2, 5, and 6 are routine computations for a computer algebra system. In the following subsections we present algorithms to automate steps 3 and 4.

3.1. Gap vectors

We first look at automating step 3 of Algorithm 3. As mentioned previously, the set of gap vectors forms an order ideal in $\mathbb{N}^{k+1}$ and therefore it suffices to find a finite basis of minimal gap vectors. In this section we present a method to test whether a given $\vec{v}$ is a gap vector by checking finitely many permutations for pattern containment.

The approach mimics that of [24], rather than [21] where Vatter presents a stronger notion of gap vector which yields an *a priori* bound on the set of vectors to check. Define the norm of a vector $\vec{v} = \langle v_1, \ldots, v_{k+1} \rangle$ to be the sum of its components, $|\vec{v}| = v_1 + \cdots + v_{k+1}$. In Vatter's notion of gap vector, if $\vec{v}$ is a basis gap vector then $|\vec{v}| \leq \max\{|\sigma| : \sigma \in B\} - 1$. In Zeilberger's method in [24], the maximum norm of basis gap vectors is entered by the user as a parameter of the algorithm.

We choose the method of [24] since its implementation allows the user more control over runtime via a parameter that lets us set the maximum allowed gap norm. This speeds computation time since it reduces the candidate pool for putative gap vectors, but this is at the cost of missing gap vectors which could make the enumeration scheme finite. For example, it is shown in [21] that there is no finite enumeration scheme for the forbidden set $B = \{\text{1-4-2-3, 1-4-3-2}\}$ using only gap vectors of norm 1. On the other hand, there is a depth 7 scheme for 1-2-3-4-5-avoiding permutations which has maximum basis gap vector norm 1 instead of the *a priori* bound of 4. A search for a depth 7 scheme with maximum gap vector norm 4 is impractical with the current implementation. A search for a depth 7 scheme with maximum gap vector norm 1, however, completes in under five minutes with a finite scheme.

Table 1
Computing gap vectors for $p = 12$ with respect to {23–1}.

$\vec{v}$	$A(12, \vec{v})$	Gap vector?
$\langle 0, 0, 0 \rangle$	{12}	No
$\langle 1, 0, 0 \rangle$	{231}	Yes
$\langle 0, 1, 0 \rangle$	{132}	No
$\langle 0, 0, 1 \rangle$	{123}	No
$\langle 1, 1, 0 \rangle$	{2413, 2431}	Yes
$\langle 1, 0, 1 \rangle$	{2314, 2341}	Yes
$\langle 0, 1, 1 \rangle$	{1324, 1342}	No
$\langle 2, 0, 0 \rangle$	{3412, 3421}	Yes
$\langle 0, 2, 0 \rangle$	{1423, 1432}	No
$\langle 0, 0, 2 \rangle$	{1234, 1243}	No

We now present a test for whether a specific vector $\vec{v}$ is a gap vector by checking finitely many cases.

Given a set of forbidden patterns B, prefix $p \in \mathcal{S}_k$, and vector $\vec{v} \in \mathbb{N}^{k+1}$, define the set of permutations with prefix p and spacing vector $\vec{v}$:

$$A(p, \vec{v}) := \{\pi \in \mathcal{S}_{|p|+|\vec{v}|} : \pi_1 \cdots \pi_k \sim p, \vec{g}(\pi_1 \cdots \pi_k) = \vec{v}\}.$$

In this notation, $\vec{v}$ is a gap vector if $\vec{u} \geq \vec{v}$ implies that every $\pi \in A(p, \vec{u})$ contains some pattern in B.

Define the *head* of a vincular pattern (σ, X) to be the subpattern $(\text{red}(\sigma_1 \cdots \sigma_{\ell+1}), X)$ where $\ell = \max X$. For example, the head of $(241652, \{2\}) = 2\text{-}41\text{-}6\text{-}5\text{-}3$ is $(\text{red}(241), \{2\}) = 2\text{-}31$. The part of σ following the head is a classical pattern, with dashes between every letter.

Theorem 4. *Consider a prefix $p \in \mathcal{S}_k$ and a spacing vector $\vec{v} \in \mathbb{N}^{k+1}$. If every permutation $\pi \in A(p, \vec{v})$ contains a copy of some $\sigma \in B$ such that $\pi_1 \cdots \pi_k$ contains the head of the copy, then $\vec{v}$ is a gap vector.*

Proof. We will demonstrate how to construct any permutation $\pi \in A(p, \vec{u})$ for $\vec{u} \geq \vec{v}$ from a $\pi' \in A(p, \vec{v})$ while preserving any copy of $\sigma \in B$ whose head lies in $\pi'_1 \cdots \pi'_k$.

Let $\pi \in A(p, \vec{u})$ where $|p| = k$. Let c_i be the ith smallest letter in $\pi_1 \cdots \pi_k$ and let $c_0 = 0$ and $c_{k+1} = n + 1$. Define $C_i := \{\pi_j : j > k, c_{i-1} < \pi_j < c_i\}$ for $i \in [k + 1]$, and observe that $u_i = |C_i|$. For each i, choose $u_i - v_i$ letters of C_i, delete these letters from π, and reduce. Note that the deleted letters all lie outside of the prefix $\pi_1 \cdots \pi_k$, so this process forms $\pi' \in A(p, \vec{v})$. Reversing this process by re-inserting the letters provides the necessary construction of π from π'. By our hypothesis, π' contains $\sigma \in B$ such that the head of σ lies in the prefix $\pi'_1 \cdots \pi'_k$. Inserting letters after the prefix will not destroy this copy of σ since the portion of σ lying outside the head has no adjacency restrictions. Hence $\pi \in A(p, \vec{u})$ also contains σ, and our result is proven. $\square$

Note that $A(p, \vec{v})$ contains $|\vec{v}|!$ permutations, and each of these must be checked for B-containment. Hence keeping $|\vec{v}|$ small is a significant computational advantage.

Note that the criterion that every permutation in $A(p, \vec{v})$ contains a copy of $\sigma \in B$ *such that $\pi_1 \cdots \pi_k$ contains the head of the copy* is required. For example, consider $B = \{124\text{-}3, 123\text{-}4\text{-}5\}$. Here $A(123, \langle 0, 0, 0, 2 \rangle) = \{12345, 12354\}$, both of which contain a forbidden pattern although the copy of 124-3 contained in 12354 does not have its head entirely in p. Now observe that $234165 \in A(123, \langle 1, 0, 0, 2 \rangle)$ avoids B even though $\langle 1, 0, 0, 2 \rangle \geq \langle 0, 0, 0, 2 \rangle$: the inserted 1 severs the copy of 124-3 without creating any other forbidden pattern.

Note that Theorem 4 provides a sufficient condition, but we do not prove necessity. There may exist gap vectors $\vec{v}$ which do not satisfy the given criterion, but we have observed no such vectors in practice.

In the computer implementation of this test, one must construct $A(p, \vec{v})$ explicitly. This may be done by methods discussed in [24].

As an example of applying Theorem 4, consider the $B = \{23\text{-}1\}$-avoiding permutations, with prefix $p = 12$ and suppose we search over all vectors with norm at most 2. Table 1 gives the relevant information for each of the ten candidates.

Looking at the set of gap vectors determined $\{\langle 1, 0, 0 \rangle, \langle 1, 1, 0 \rangle, \langle 1, 0, 1 \rangle, \langle 2, 0, 0 \rangle\}$, we see the order ideal generated by these vectors has minimal basis $\{\langle 1, 0, 0 \rangle\}$.[3] Hence in the enumeration scheme we see $G_{12} = \{\langle 1, 0, 0 \rangle\}$.

3.2. Reversibly deletable sets

We now turn our attention to automating step 4 of Algorithm 3: discovering reversibly deletable sets of indices for a given prefix p. Our scenarios-based approach parallels that of [24].

Recall that for a set of indices R, the map d_R deletes π_r for each $r \in R$. This forms a bijection $d_R : \mathcal{S}_n(\emptyset)[p; w] \to \mathcal{S}_{n-|R|}(\emptyset)[d_R(p); d_R(w)]$, and when this map restricts to a bijection $\mathcal{S}_n(B)[p; w] \to \mathcal{S}_{n-|R|}(B)[d_R(p); d_R(w)]$ we say that R

---

[3] The thoroughness of this example is perhaps misleading regarding the implementation. Once the computer discovers that $\vec{v}$ is a gap vector, it need not bother testing any other $\vec{u} \geq \vec{v}$.

Table 2
$d_R(\pi)$ for each $R \subseteq \{1, 2, 3\}$, $\pi \in A_{132}$.

$\pi \in A_{132}$	13246•5	14236•5	15236•4	16235•4
$d_{\{1\}}(\pi)$	2135•4	3125•4	4125•3	5124•3
$d_{\{2\}}(\pi)$	1235•4	1235•4	1235•4	1235•4
$d_{\{3\}}(\pi)$	1235•4	1325•4	1425•3	1524•3
$d_{\{1,2\}}(\pi)$	124•3	124•3	124•3	124•3
$d_{\{1,3\}}(\pi)$	124•3	214•3	314•2	413•2
$d_{\{2,3\}}(\pi)$	124•3	124•3	134•2	143•2
$d_{\{1,2,3\}}(\pi)$	13•2	13•2	13•2	13•2

is *reversibly deletable* for the prefix p. In the classical case, the deletion of a letter or letters could not create a copy of a forbidden pattern. For vincular patterns, however, deleting a letter may create the adjacency required to form an occurrence of a vincular pattern. For example, 3142 avoids 23-1 but $d_2(3142) = 231$ does not since the 3 and 4 become adjacent to one another. This does not preclude the existence of bijective maps d_R, it merely requires additional checks for the automated discovery. In the end, a finite search for a reversibly deletable set suffices as in the classical case: it is only the manner in which we check each candidate which differs. The need to check both directions of the map first appears in [18] when extending schemes for barred pattern avoidance. The added twist needed for vincular pattern avoidance is the introduction of the "null" symbol •.

Note R is reversibly deletable when every $\pi \in \mathcal{S}_n(\emptyset)[p; w]$ avoids B if and only $d_R(\pi)$ also avoids B. Inversely, we could check whether every π which contains some $\sigma \in B$ has image $d_R(\pi)$ which also contains some $\sigma' \in B$. This approach was introduced by Zeilberger in [24] and used by the second author in [16,18] when extending enumeration schemes to other contexts.

Let us illustrate the approach via an example before moving to the general case. Consider $B = \{124\text{-}3\}$ and prefix $p = 132$. We ask "which letters of the prefix can participate in a $\sigma = 124\text{-}3$ pattern?". Suppose π is a permutation with prefix pattern 132, and that at least one letter of the prefix is part of a copy of σ. If π has minimal length, then π must have the form red($132abc$) where $a, b, c \in \mathbb{Q}$ such that $2abc \sim \sigma$: σ starts with two rises and the descent 32 in the prefix prevents a σ from starting earlier. There are four such permutations: 13$\underline{2465}$, 14$\underline{2365}$, 15$\underline{2364}$, 16$\underline{2354}$ (the occurrence of σ is underlined in each). It will be necessary to keep track of where and when dashes in the contained copy of σ appear outside of the prefix, which we denote with the "null" symbol •. This special character denotes the possibility for intervening letters but cannot participate in patterns itself. Thus we write these four permutations as 13$\underline{246}$•$\underline{5}$, 14$\underline{236}$•$\underline{5}$, 15$\underline{236}$•$\underline{4}$, 16$\underline{235}$•$\underline{4}$. Denote this set of *containment scenarios* for $p = 132$ by A_{132}. We now apply d_R for each $R \subseteq [3]$ and check whether the images under d_R each contain σ. This is done in Table 2. If $d_R(\pi)$ contains σ for each $\pi \in A_{132}$, then R passes the first test[4] for reverse deletability: the insertion d_R^{-1} does not create any forbidden patterns in B when applied to a permutation which already avoids B. Looking across the rows of Table 2, we see that $\{1\}$, $\{2\}$, and $\{1, 2\}$ pass this test since every permutation in those rows contains σ.

Since a deletion map creates new adjacencies and potentially a copy of σ, there is a second test that R must pass to be reversibly deletable. Consider $R = \{2\}$, which passed the previous test. Applying $d_{\{2\}}$ to a permutation with prefix pattern 132 will create a permutation with prefix pattern 12, so we must consider the containment scenarios for the prefix $p = 12$: $A_{12} = \{124\text{•}3, 1235\text{•}4\}$. We then consider all ways each of these containment scenarios could have arisen by applying $d_{\{2\}}$ to a permutation with prefix pattern 132; i.e., every permutation of the form red($1a24$•3) for $a \in \{2 + \frac{1}{2}, 3 + \frac{1}{2}, 4 + \frac{1}{2}\}$ or red($1b235$•4) for $b \in \{2 + \frac{1}{2}, 3 + \frac{1}{2}, 4 + \frac{1}{2}, 5 + \frac{1}{2}\}$. In particular, this list includes 1325•4, which avoids σ while $d_{\{2\}}(1325$•$4) = 124$•3 contains σ. Since one can use $d_{\{2\}}$ to create a σ-containing permutation from a σ-avoiding permutation, $R = \{2\}$ cannot be reversibly deletable. On the other hand, for $R = \{1\}$ one can check that the containment scenarios for $d_{\{1\}}(132) = 21$ are $A_{21} = \{2135$•$4, 3125$•$4, 4125$•$3, 5124$•$3\}$ and that the permutations starting with 132 which map to some $\pi \in A_{21}$ are precisely $\{13246$•$5, 14236$•$5, 15236$•$4, 16235$•$4\}$. Since each of these pre-image permutations contains σ, $R = \{1\}$ passes the second test for reversible deletability. Hence $\{1\}$ is reversibly deletable. Similarly, for $R = \{1, 2\}$ we get only the containment scenario $A_1 = \{124$•$3\}$ and the same set of pre-images with prefix 132:

$$d_{\{1,2\}}(\{13246\text{•}5, 14236\text{•}5, 15236\text{•}4, 16235\text{•}4\}) = A_1.$$

Again, each of the permutations on the lefthand side contains σ, so $R = \{1, 2\}$ is reversibly deletable. Hence we have two non-empty reversibly deletable sets for prefix 132. While either set will lead to a valid enumeration scheme, we follow a convention to choose the largest one and break ties lexicographically by the smallest elements.

To demonstrate a subtlety of containment scenarios, consider the forbidden set $B = \{3\text{-}21, 32\text{-}1\}$ and prefix $p = 21$. Here we see that we have the basis gap vector $\langle 1, 0, 0 \rangle$, and so any permutation starting with prefix word ab for $a > b > 1$ necessarily contains a forbidden pattern. Hence to prove R is reversibly deletable, we only need to show d_R is bijective starting from sets of the form $\mathcal{S}_n(B)[21; a1]$. Therefore even though 42•31 contains a forbidden pattern and begins with 21, we know

---

[4] In the classical case, this was the *only* test.

that $\mathcal{S}_n(B)[21; 42] = \emptyset$ and so we do not need to check whether $d_R(42\bullet31)$ contains a forbidden pattern. In fact the only containment scenario worth checking for $p = 21$ is $41\bullet32$. Hence $R = \{2\}$ passes the first test for reversible deletability. We then move on to consider the containment scenarios for prefix pattern $d_2(21) = 1$. These are $A_1 = \{3\bullet21, 32\bullet1\}$. The pre-images under d_2 starting with 21 include 413-2, however, which does not contain either forbidden pattern. Hence $R = \{2\}$ fails the second test for reversible deletability. If we had not kept track of dashes with the null character $\bullet$, however, the preimage 4132 would have contained a forbidden pattern and $\{2\}$ would have appeared to be reversibly deletable.

We now outline in general the scenario method to test whether a set R is reversibly deletable for prefix p with respect to forbidden pattern B. We begin with a formal definition for a containment scenario.

Definition 5. Let (σ, X) be a vincular pattern of length ℓ and let $p \in \mathcal{S}_k$ be a prefix pattern with known set of gap vectors G. Let $w \in \{1, 2, \ldots, n, \bullet\}^{n+m}$ be a word with m copies of $\bullet$ and no other letters repeated. Then w is a *containment scenario* for p if the following criteria are satisfied:

1. $w_1 \cdots w_k \sim p$. Note this implies $\bullet$ does not appear in the first k letters.
2. $w_1 \cdots w_k$ fails all gap vector criteria in G.
3. There is some subsequence $1 \le i_1 < \cdots < i_\ell \le n + m$ such that $w(i_1) \cdots w(i_\ell) \sim \sigma$ and $w(i_x + 1) = \bullet$ for each x such that $i_x \ge k$ and there is a dash between σ_x and σ_{x+1} (i.e., $x \notin X$).
4. No subsequence of w is a containment scenario.

The set of containment scenarios for a forbidden set B is simply the union of the sets of containment scenarios for each $\sigma \in B$. We will denote by A_p the set of containment scenarios for a forbidden set B, a prefix p, and a set of gap vectors G.

One can compute A_p via brute force over all $2^{|p|} - 1$ nonempty subsequences of p. A set of indices $1 \le i_1 < \cdots i_t \le |p|$ is a *partial match* for $(\sigma, X) \in B$ if $i_x + 1 = i_{x+1}$ for each $x \in X$ and $p_{i_1} p_{i_2} \cdots p_{i_t} \sim \sigma_1 \sigma_2 \cdots \sigma_t$. Note that a set of indices may be a partial match for more than one pattern in B. For each partial match of (σ, X), insert the $|\sigma| - t$ letters and necessary number of $\bullet$ on the right end of p in such a way to complete the occurrence of σ using the letters in the partial match. Repeating this process for each $(\sigma, X) \in B$ gives us the complete set of containment scenarios. We may then throw out any containment scenarios whose first $|p|$ letters satisfy a gap vector criterion for some basis gap vector.

We now present the algorithm to check whether a given set $R \subseteq [k]$ is reversibly deletable for a prefix $p \in \mathcal{S}_k$ with respect to the forbidden set B.

Algorithm 6. 1. Compute the set of containment scenarios A_p.
2. For each $\pi \in A_p$, check if $d_R(\pi)$ contains a forbidden pattern in B. If any π avoids B, then R is not reversibly deletable. Otherwise, proceed to step 3.
3. Compute the set of containment scenarios $A_{d_R(p)}$.
4. Find the set of all permutations π with prefix p such that $d_R(\pi) \in A_{d_R(p)}$. If any of these π avoids B, then R is not reversibly deletable. If each of these contains some forbidden pattern, then R is reversibly deletable.

Therefore we can compute non-empty reversibly deletable sets automatically by a finite computer search. This concludes our discussion on automated discovery of enumeration schemes. These procedures have been implemented in the Maple package GVATTER, available on the authors' homepages.

4. Special cases of guaranteed success

Knowing *a priori* whether a set of patterns B has a finite enumeration scheme remains an open question. As a partial classification, we show here that if B contains only consecutive (i.e., dashless) patterns and patterns of the form $\sigma_1\sigma_2 \cdots \sigma_{t-1}$-$\sigma_t = (\sigma, [t-2])$ then B admits a finite enumeration scheme. We will prove this via a series of lemmas regarding gap vectors and reversibly deletable sets.

Our first sequence of lemmas regards consecutive pattern avoidance.

Lemma 7. *Let $\sigma \in \mathcal{S}_t$ and consider prefix pattern $p \in \mathcal{S}_k$. Then the set $\{1\}$ is reversibly deletable for prefix p with respect to $\{(\sigma, [t-1])\}$ if $p_1p_2 \cdots p_{\min(t,k)} \not\sim \sigma_1\sigma_2 \cdots \sigma_{\min(t,k)}$.*

Proof. Suppose the permutation starts with prefix pattern p such that $\pi_1 p_1p_2 \cdots p_{\min(t,k)} \not\sim \sigma_1\sigma_2 \cdots \sigma_{\min(t,k)}$. Then π_1 could not be involved in a copy of the consecutive pattern $(\sigma, [t-1])$, since otherwise $p_1p_2 \cdots p_{\min(t,k)} \sim \sigma_1\sigma_2 \cdots \sigma_{\min(t,k)}$. To verify $\{1\}$ is reversibly deletable, observe that applying the deletion map d_1 does not create new adjacencies, and so $d_1(\pi)$ avoids $(\sigma, [t-1])$ if and only if π avoids $(\sigma, [t-1])$. $\square$

Lemma 7 is conservative since larger sets can be reversibly deletable. The set $\{1, 2, \ldots, s\}$ is reversibly deletable for prefix p with respect to $(\sigma, [t-1])$ if $p_a p_{a+1} \cdots p_{\min(a+t-1,|p|)}$ is not order-isomorphic to $\sigma_1\sigma_2 \cdots \sigma_{\min(t,|p|-a+1)}$ for each $a \le s$. This fact can be proven by the same arguments, since $d_{\{1,2,\ldots,s\}}$ will not create new adjacencies. Looking ahead to Lemma 14, we will keep to $s = 1$ so that the only non-empty reversibly deletable set which appears in a given scheme is $\{1\}$.

Lemma 8. *Let $\sigma \in \mathcal{S}_t$. For any permutation p containing the consecutive pattern $(\sigma, [t-1])$, $\vec{0} = \langle 0, 0, \ldots, 0 \rangle$ is a gap vector for prefix p with respect to $\{(\sigma, [t-1])\}$.*

Proof. If p already contains σ, then no permutation containing p (in particular, starting with prefix pattern p) can avoid σ. Hence $\mathcal{S}_n((\sigma, [t-1]))[p; w] = \emptyset$ for any prefix word w, and the result follows from the definition of gap vectors. $\square$

Lemmas 7 and 8 combine to give us the following proposition.

Proposition 9. *For $\sigma \in \mathcal{S}_t$, the pattern set $\{(\sigma, [t-1])\}$ admits an enumeration scheme of depth t where every reversibly deletable set is either $\emptyset$ or $\{1\}$.*

Proof. We will construct an enumeration scheme E for $(\sigma, [t-1])$-avoiding permutations. For each permutation p of length at most t, Lemmas 7 and 8 imply a gap vector basis G_p and a reversibly deletable set R_p, and we add the triple (p, G_p, R_p) to E. Observe every permutation of length exactly t is either $p = \sigma$, in which case $\vec{0}$ is a gap vector, or $p \neq \sigma$, in which case $\{1\}$ is reversibly deletable, so we see that E satisfies criterion (2a) given in Section 2.3. Since every permutation p of length at most t appears in E, we see that criterion (2b) is also satisfied. Hence E is a valid enumeration scheme with finitely many elements. $\square$

This proof has some redundancy built into it. For a more efficient approach, we could execute Algorithm 3, using Lemmas 7 and 8 to obtain gap vector bases (either $\emptyset$ or $\{\vec{0}\}$) and our reversibly deletable sets (either $\emptyset$ or $\{1\}$). The algorithm terminates since every prefix pattern p of length t is either $p = \sigma$, in which case $\vec{0}$ is a gap vector, or $p \neq \sigma$, in which case $\{1\}$ is reversibly deletable. Hence E contains no prefix patterns of length greater than t.

We now present the analogues of Lemmas 7 and 8 and Proposition 9 for patterns of the form $(\sigma, [t-2])$ for $\sigma \in \mathcal{S}_t$.

Lemma 10. *Let $\sigma \in \mathcal{S}_t$ and consider prefix pattern $p \in \mathcal{S}_k$. Then the set $\{1\}$ is reversibly deletable for prefix p with respect to $(\sigma, [t-2])$ if $p_1 p_2 \cdots p_{\min(t-1,k)} \not\sim \sigma_1 \sigma_2 \cdots \sigma_{\min(t-1,k)}$.*

Proof. As in the proof for 7, the letter π_1 cannot be involved in the pattern $(\sigma, [t-2])$ unless $\pi_1 \pi_2 \cdots \pi_{\min(t-1,k)}$ is order-isomorphic to $\sigma_1 \sigma_2 \cdots \sigma_{\min(t-1,k)}$. Furthermore, the deletion map d_1 does not create new adjacencies and so preserves $(\sigma, [t-2])$-avoidance.

Like Lemma 7, Lemma 10 is conservative. The set $\{1, 2, \ldots, s\}$ is reversibly deletable for prefix p with respect to $(\sigma, [t-2])$ if $p_a p_{a+1} \cdots p_{\min(a+t-2,|p|)}$ is not order-isomorphic to $\sigma_1 \sigma_2 \cdots \sigma_{\min(t-1,|p|-a+1)}$ for each $a \leq s$.

Lemma 11. *Let $\sigma \in \mathcal{S}_t$ and consider the prefix pattern $p = \mathrm{red}(\sigma_1 \cdots \sigma_{t-1})$. Then $\vec{v} = \langle 0, \ldots, 0, 1, 0, \ldots, 0 \rangle$ is a gap vector for the prefix p with respect to $\{(\sigma, [t-2])\}$, where $v_{\sigma_t} = 1$ and $v_i = 0$ for other i. Further, $\{1\}$ is reversibly deletable for p with respect to $(\sigma, [t-2])$.*

Proof. Let $\pi \in A(p, \vec{u})$ for $\vec{u} \geq \vec{v}$. Then we know $\pi_1 \cdots \pi_{t-1} \sim \sigma_1 \cdots \sigma_{t-1}$, so define indices a and b so that π_a corresponds to $\sigma_t - 1$ and π_b corresponds to $\sigma_t + 1$. Since π has spacing vector $\vec{u} \geq \vec{v}$, we know there is some π_i for $i > t-1$ such that the value π_i lies between the values of π_a and π_b. In short, $\pi_1 \cdots \pi_{t-1} \pi_i$ forms a copy of $(\sigma, [t-2])$. Hence $\vec{v}$ is a gap vector as per Theorem 4.

We now turn to proving $\{1\}$ is reversibly deletable. Suppose that $\pi \in \mathcal{S}_n((\sigma, [t-2]))[p; w]$ for a prefix word w such that $\vec{u} = \vec{g}(n, w) \not\geq v$. Then $u_{\sigma_t} = 0$, so we see that there is no π_i for $i > t-1$ such that $\pi_1 \cdots \pi_{t-1} \pi_i \sim \sigma$. Therefore π_1 cannot be involved in any copies of $(\sigma_1 \sigma_2 \cdots \sigma_{t-1}, [t-2])$ and hence cannot be involved in a copy of $(\sigma, [t-2])$. Since d_1 preserves $(\sigma, [t-2])$-avoidance as previously seen, we have shown that $\{1\}$ is reversibly deletable. $\square$

We combine Lemmas 10 and 11 to form Proposition 12.

Proposition 12. *For $\sigma \in \mathcal{S}_t$, the pattern set $\{(\sigma, [t-2])\}$ admits an enumeration scheme of depth $t-1$ where every reversibly deletable set is either $\emptyset$ or $\{1\}$.*

Proof. The proof is similar to that of Proposition 9. Construct E by adding the triple (p, G_p, R_p) for each permutation p of length at most $t-1$, where the gap vector basis G_p and reversibly deletable set R_p are given by Lemmas 10 and 11. Since every permutation p of length $t-1$ has $R_p = \{1\}$, we see that E is finite while still satisfying criterion (2a). $\square$

As in Proposition 9, the scheme constructed in Proposition 12 has more terms than the scheme which would be constructed via Algorithm 3.

Propositions 9 and 12 demonstrate how to construct schemes for singleton sets of patterns of certain forms. The following lemmas outline when one can combine enumeration schemes for pattern sets B and B' to construct an enumeration scheme for $B \cup B'$.

Lemma 13. *Suppose that $\vec{v}$ is a gap vector for the prefix p with respect to B. Then $\vec{v}$ is a gap vector for p with respect to any pattern set $C \supseteq B$. In particular, for a pattern set B', $\vec{v}$ is a gap vector for $C = B \cup B'$.*

Proof. If $C \supseteq B$, then $\mathcal{S}_n(C)[p; w] \subseteq \mathcal{S}_n(B)[p; w]$. Therefore any criterion implying $\mathcal{S}_n(B)[p; w] = \emptyset$, in particular a gap vector criterion, also implies that $\mathcal{S}_n(C)[p; w] = \emptyset$. $\square$

Lemma 14. *Suppose that R is reversibly deletable for the prefix p with respect to B and with respect to B'. Then R is reversibly deletable for p with respect to $B \cup B'$.*

Proof. First recall that $\mathcal{S}_n(B \cup B')[p; w] = \mathcal{S}_n(B)[p; w] \cap \mathcal{S}_n(B')[p; w]$. From the definition of a reversibly deletable set, we know that we have the following bijections:

$$d_R : \mathcal{S}_n(B)[p; w] \to \mathcal{S}_{n-|R|}(B)[d_R(p); d_R(w)]$$
$$d_R : \mathcal{S}_n(B')[p; w] \to \mathcal{S}_{n-|R|}(B')[d_R(p); d_R(w)].$$

It follows that we also have the bijection

$$d_R : \mathcal{S}_n(B \cup B')[p; w] \to \mathcal{S}_{n-|R|}(B \cup B')[d_R(p); d_R(w)].$$

Hence R is reversibly deletable for p with respect to $B \cup B'$. □

We may combine Lemmas 13 and 14 combine with Propositions 9 and 12 to give us the following theorem.

Theorem 15. *If a finite set B contains only patterns of the form $(\sigma, [|\sigma| - 1])$ and $(\sigma, [|\sigma| - 2])$, then B admits a finite enumeration scheme where every reversibly deletable set is either $\emptyset$ or $\{1\}$.*

Proof. Let M be the maximum length of patterns $(\sigma, X) \in B$. For each permutation p of length at most M, let $G_p(\sigma, X)$ be the gap vector basis for prefix pattern p with respect to (σ, X) as implied by Lemmas 8 or 11. Define the set

$$G_p := \bigcup_{(\sigma, X) \in B} G_p(\sigma, X). \tag{3}$$

By Lemma 13, G_p is a gap vector basis for p with respect to B. Observe that G_p may not be a minimal basis.

Similarly let $R_p(\sigma, X)$ be the reversibly deletable set for p with respect to (σ, X) as implied by Lemmas 7, 10, or 11. Define the set

$$R_p := \bigcap_{(\sigma, X) \in B} R_p(\sigma, X). \tag{4}$$

Observe that each $R_p(\sigma, X)$ is either $\emptyset$ or $\{1\}$, so R_p is either $\emptyset$ or $\{1\}$. Further, $R_p = \{1\}$ if and only if each $R_p(\sigma, X) = \{1\}$. Hence Lemma 14 implies R_p is reversibly deletable for p with respect to B.

Let E be the set of triples (p, G_p, R_p) for each permutation p of length at most M. Clearly E is finite and satisfies criteria (1) and (2b), and E satisfies (2a) for any prefix p of length less than M. It remains to show that any prefix p of length M has either $\vec{0} \in G_p$ or non-empty R_p. If p contains any forbidden pattern $(\sigma, X) \in B$, then $G_p(\sigma, X) = \{\vec{0}\}$ by Lemmas 8 or 11. Hence $\vec{0} \in G_p$. On the other hand if p avoids B, then p avoids each $(\sigma, X) \in B$ and so each $R_p(\sigma, X) = \{1\}$ by Lemmas 7, 10, or 11. Hence $R_p = \{1\}$. Thus we see that E satisfies criterion (2a) for all p of length M, and so E is a valid enumeration scheme for B. □

It is worth mentioning that Proposition 12 and its supporting lemmas can be generalized to obtain a finite scheme for any set of patterns of the form $B = \{\sigma_1 \sigma_2 \cdots \sigma_t \text{-} \sigma_{t+1} \text{-} \sigma_{t+2}, \sigma_1 \sigma_2 \cdots \sigma_t \text{-} \sigma_{t+2} \text{-} \sigma_{t+1}\}$ based on the enumeration scheme for their common head $\tau = \text{red}(\sigma_1 \sigma_2 \cdots \sigma_t)$. In this case the prefix $p = \tau$ has a gap vector basis $G_p = \{\vec{v}\}$, where $v_{\sigma_{t+1}} = 1$, $v_{\sigma_{t+2}-1} = 1$ and $v_i = 0$ for other i for $\sigma_{t+1} < \sigma_{t+2} - 1$. If $\sigma_{t+1} = \sigma_{t+2} - 1$, then $v_{\sigma_{t+1}} = v_{\sigma_{t+2}-1} = 2$ instead. The triples for other prefixes $p \neq \tau$ appearing in E_τ transfer unchanged to E_σ.

One may similarly construct a scheme for the set of $k!$ patterns formed by appending to the consecutive pattern τ a dashed tail of k letters $\sigma_{t+1}, \sigma_{t+1}+1, \ldots, \sigma_{t+k}$ in all possible orderings, e.g., $B = \{12\text{-}3\text{-}4\text{-}5, 12\text{-}3\text{-}5\text{-}4, 12\text{-}4\text{-}3\text{-}5, 12\text{-}4\text{-}5\text{-}3, 12\text{-}5\text{-}3\text{-}4, 12\text{-}5\text{-}4\text{-}3\}$. If we suppose $\sigma_{t+j} < \sigma_{t+j+1}$, then let $\vec{u}^{(j)}$ be the 0-1 vector with a 1 in position $\sigma_{t+j} - (j-1)$. Then for prefix $p = \tau$ we have the gap vector $\vec{v} = \sum_j \vec{u}^{(j)}$. The remaining triples $(p, G_p, R_p) \in E_\tau$ transfer unchanged to E_σ. These pattern sets are identical to Kitaev's notion of partially-ordered generalized patterns in [13], where some letters of the pattern are incomparable (or rather, do not need to be compared). Thus the example B would be written as a single such pattern $12\text{-}3\text{-}3'\text{-}3''$ where the letters acting as $3, 3', 3''$ are incomparable. The arguments above imply that the following is an enumeration scheme for patterns avoiding $12\text{-}3\text{-}3'\text{-}3''$:

$$\{(\epsilon, \emptyset, \emptyset), (1, \emptyset, \emptyset), (12, \{\langle 0, 0, 3 \rangle\}, \{1\}), (21, \emptyset, \{1\})\}. \tag{5}$$

One might hope we can continue this trend of adding dashed portions to patterns with known schemes to get new schemes, but of course this does not work in general.[5] Still, there may be some interesting relationships between two vincular patterns with the same underlying permutation. For example, every pattern $(1234, X)$ for $X \subseteq \{1, 2, 3\}$ has a finite scheme, whose depths (based on the implementation in GVATTER) are summarized in Table 3. There do not appear to be any clear patterns dictating scheme depth for the vincular pattern $(1234, X)$ given the depths of other patterns $(1234, X')$, based on subset relations between X and X'.

---

[5] If it did then we would get enumeration schemes for all classical patterns, which certainly is not the case.

Table 3
Scheme depth for vincular pattern $(1234, X)$.

σ	X	Scheme depth
1234	$\{1, 2, 3\}$	4
123-4	$\{1, 2\}$	3
12-34	$\{1, 3\}$	4
1-234	$\{2, 3\}$	4
12-3-4	$\{1\}$	4
1-23-4	$\{2\}$	3
1-2-34	$\{3\}$	5
1-2-3-4	$\emptyset$	4

Table 4
Success rate by block type.

Block type	Number of trivial symmetry classes	Number of $(5, 2)$-SC classes	Percentage (%)
(2)	1	1	100
(1,1)	1	1	100
(3)	2	2	100
(2,1)	3	3	100
(1,1,1)	2	2	100
(4)	8	8	100
(3,1)	12	12	100
(2,2)	8	3	37.5
(2,1,1)	12	4	25
(1,2,1)	8	6	75
(1,1,1,1)	7	2	28.6

5. Known successes and applications

5.1. Analysis of success rates

Aside from the results of the previous section, there is no known classification of which pattern sets admit finite enumeration schemes. In this section we present empirical results obtained from the implementation of the above algorithms in the Maple package GVATTER. We will say that a set of forbidden patterns B is (d, M)-*scheme countable*, or (d, M)-SC, if either B, B^r, or B^{-1} admits a finite enumeration scheme of depth at most d with basis gap vectors with norm at most M. As discussed in Section 2, B is (d, M)-SC if and only if its set of complement patterns B^c is (d, M)-SC.

The following data were assembled by checking whether each vincular pattern (σ, X) is $(5, 2)$-SC, where we chose 5 and 2 as practical computational considerations. While there are $k! \cdot 2^{k-1}$ vincular patterns of length k, we took advantage of symmetry when able to reduce the number of patterns to check. To refine analysis, we separated the patterns of length k by the locations of their dashes. These are represented in Table 4 by "block type," which is a vector describing the number of letters between each dash. For example, the block type of the pattern 12-35-467 is $(2, 2, 3)$.

It would appear that the success rate is not solely dependent on the number of dashes. For example, of the 20 pattern classes with a single dash, the five which are not $(5, 2)$-SC are all of block type $(2, 2)$. Of the classes with two or more dashes, the most successful block type is $(1, 2, 1)$ where the dashes do not follow one another.

In the classical case, schemes were most successful when avoiding multiple patterns simultaneously. Table 5 lists the success rates for finding sets B which are $(5, 2)$-SC, for various $B \subseteq \mathcal{S}_2 \cup \mathcal{S}_3 \cup \mathcal{S}_4 \cup \mathcal{S}_5$. In the leftmost column, the "set type" of a set B refers to the multiset $\{|\sigma| : \sigma \in B\}$.

5.2. Wilf-classification of vincular patterns

We now present some preliminary Wilf-classification results based on the data generated by the schemes. Two patterns σ, τ are said to be *Wilf-equivalent* if $s_n(\sigma) = s_n(\tau)$ for all n, and we denote this $\sigma \equiv \tau$. Claesson enumerates permutations avoiding a length 3 pattern with one dash in [7], and Elizalde and Noy enumerate permutations avoiding length 3 patterns with no dashes (i.e., consecutive) in [10]. Thus we turn our attention to length 4 patterns. All patterns of length 4 with finite schemes of depth at most 5 are listed in Table 6. Solid black lines separate classes whose sequences are observed to diverge before the 31st term.

The first steps towards general results were taken in [9,13], which we summarize below in Proposition 16:

Proposition 16 (*Elizalde [9], Kitaev [13]*)**.** *Suppose σ, τ are Wilf-equivalent consecutive patterns of length k. Then the following are also Wilf-equivalent:*

- $\sigma\text{-}(k + 1) \equiv \tau\text{-}(k + 1)$

Table 5
Success rate for sets of patterns, *B*.

Set type	Number of trivial symmetry classes	Number of (5, 2)-SC classes	Percentage (%)
{2}	2	2	100.0
{2, 2}	3	3	100.0
{2, 3}	11	11	100.0
{3}	7	7	100.0
{3, 3}	70	68	97.1
{3, 3, 3}	358	354	98.9
{4}	55	35	63.6
{4, 4}	4624	1600	34.6
{5}	479	144	30.1
{3, 4}	914	639	69.9
{3, 5}	7411	2465	33.3

Table 6
Dashed patterns of length 4 admitting schemes of depth at most 5.

σ	$\{S_n(\sigma)\}_n$	OEIS [15]	Comments
123-4	1, 2, 6, 23, 108, 598, 3815, 27532, 221708, 197025, …	A071076	
321-4			$\equiv$123-4 by Proposition 16
132-4	1, 2, 6, 23, 107, 585, 3671, 25986, 204738, 1776327, …	A071075	
231-4			$\equiv$132-4 by Proposition 16
312-4			$\equiv$132-4 by Proposition 16
213-4			$\equiv$132-4 by Proposition 16
142-3			$\equiv$132-4 by Conjecture 17
241-3			$\equiv$132-4 by Conjecture 17
124-3	1, 2, 6, 23, 107, 584, 3660, 25910, 204564, 1782520, …	New	
421-3			$\equiv$124-3 by Conjecture 17
143-2	1, 2, 6, 23, 107, 582, 3622, 25369, 197523, 1692535, …	New	
214-3	1, 2, 6, 23, 107, 583, 3637, 25548, 199506, 1714383, …	New	
12-34	1, 2, 6, 23, 107, 585, 3669, 25932, 203768, 1761109, …	A113226	
12-43			$\equiv$12-34 by Proposition 16
21-43			$\equiv$12-43 by Proposition 16
1-24-3	1, 2, 6, 23, 104, 532, 3004, 18426, 121393, 851810, …	A137538	Wilf-equivalent to 25$\bar{1}$34
1-42-3			$\equiv$1-24-3 by Conjecture 17
1-23-4	1, 2, 6, 23, 105, 549, 3207, 20577, 143239, 1071704, …	A113227	
1-32-4			$\equiv$1-23-4 by Proposition 16
1-34-2			$\equiv$1-23-4 by Conjecture 17
1-43-2			$\equiv$1-23-4 by Conjecture 17
12-3-4	1, 2, 6, 23, 105, 550, 3228, 20878, 146994, 1116000, …	New	
12-4-3			$\equiv$12-3-4 by Conjecture 17
21-3-4			$\equiv$12-3-4 by Conjecture 17
21-4-3			$\equiv$12-3-4 by Conjecture 17

- σ-$(k+2)(k+1) \equiv \tau$-$(k+2)(k+1)$
- σ-$(k+2)(k+1) \equiv \sigma$-$(k+1)(k+2)$
- 1-$(\sigma_1+1)(\sigma_2+1)\cdots(\sigma_k+1)$-$(k+1) \equiv$ 1-$(\tau_1+1)(\tau_2+1)\cdots(\tau_k+1)$-$(k+1)$.

It is clear from symmetries of the square that the following consecutive patterns are Wilf-equivalent:

- $12 \equiv 21$
- $123 \equiv 321$
- $132 \equiv 213 \equiv 231 \equiv 312$.

Thus we see that Proposition 16 gets us many of the equivalences which appear in Table 6. There remain many conjectured pairs, which we summarize below in Conjecture 17. Note that in each case, the conjectured equivalence is confirmed computationally for permutations of length $n \leq 30$.

Conjecture 17. *We have the following Wilf-equivalences:*

(a) 132-4 $\equiv$ 142-3 $\equiv$ 241-3
(b) 124-3 $\equiv$ 421-3
(c) 12-3-4 $\equiv$ 12-4-3 $\equiv$ 21-3-4 $\equiv$ 21-4-3

Table 7
Number of permutations avoiding 1–32 of length n with k inversions.

n	$k = 0$	1	2	3	4	5	6	7	8	9	10
0	1										
1	1										
2	1	1									
3	1	1	2	1							
4	1	1	2	4	3	3	1				
5	1	1	2	4	7	8	9	9	6	4	1

(d) 1-24-3 $\equiv$ 1-42-3
(e) 1-43-2 $\equiv$ 1-23-4 $\equiv$ 1-34-2 $\equiv$ 1-32-4.

The equivalences in (a) and (b) are proven in an upcoming paper by the first author [4]. The equivalences in (c) are proven in a different upcoming paper by the first author [5].

5.3. Refinement according to the inversion number

In [3], the first author demonstrates that schemes for permutations avoiding classical patterns can be adapted to compute the number of such permutations with k inversions. The same refinement applies to the schemes developed above, since they use the same deletion maps d_R.

As an example, consider the permutations avoiding 1-32. In Proposition 3 of [7], Claesson shows that the Bell numbers enumerate 1-32-avoiding permutations via a bijection, where the number of descents in the permutation is one more than the number of blocks in the corresponding set partition. We now find the number of permutations with k inversions avoiding 1-32, leading to a new refinement of the Bell numbers which is shown in Table 7. Further terms of the sequence can be generated via the enumeration scheme for 1-32 and applying the refinements discussed in [3].[6]

It is interesting to note that if one continues this chart, the columns are each eventually constant. Specifically, if $f(n, k)$ is the number of permutations avoiding 1-32 of length n with k inversions, then $f(n, k) = f(k + 1, k)$ for all $n \geq k + 1$. The stagnation can be seen as follows. Suppose π is a permutation with k inversions and length $n \geq k + 2$ which avoids 1-32. We will show that π must have $\pi_j = j$ for $j \geq k + 2$, which implies $f(n, k) = f(k + 1, k)$ for $n \geq k + 1$. First observe that π must end with an ascending run, that is, $\pi_{k+1} < \pi_{k+2} < \cdots < \pi_n$. To see this, suppose for contradiction that $\pi_j > \pi_{j+1}$ for $j \geq k + 1$. Then $\pi_i > \pi_{j+1}$ for each $i < j$ or else $\pi_i \pi_j \pi_{j+1}$ forms a copy of 1-32, but this implies π_{j+1} is involved in $j \geq k + 1$ inversions which contradicts the assumption that π has only k inversions in total. Next suppose $\pi_j < j$ for some $j \geq k + 2$, and without loss of generality assume this is the minimal such j. Then $\pi_{j-1} > \pi_j$, so $\pi_j = 1$ or else $1\pi_{j-1}\pi_j$ forms a copy of 1-32. This implies that π_j is involved in $j - 1 \geq k + 1$ inversions, however, which contradicts our assumption. Thus we have shown for each $k + 2 \leq j \leq n$ that $j \leq \pi_j < \pi_{j+1}$, and so it follows that $\pi_j = j$.

The sequence of $\{\lim_{n \to \infty} f(n, k)\}_{k \geq 0}$ is given by 1, 1, 2, 4, 7, 13, 22, 38, 63, 105, ..., which was new to the OEIS [15] and has been added as A188920. These numbers ought to describe set partitions where a certain statistic is k, but it is unclear whether such a statistic would be "natural." It is interesting to note that when one considers the analogous question of permutations avoiding 1-3-2 of length n and k inversions, one also sees that as $n \to \infty$ the number is eventually constant at the number of integer partitions of k. See [11] for a proof in terms of lattice paths or [8] for a more recent treatment.

6. Conclusions and future directions

This paper develops automatable methods to compute $s_n(B)$ for many sets of vincular patterns B. This was accomplished by extending the enumeration schemes developed by Vatter and Zeilberger in [21,23,24]. The restrictions on adjacencies which vincular patterns present introduced complications when discovering gap vectors and reversibly deletable sets. Theorem 4 demonstrates that gap vectors can only be discovered when prefixes are long enough to contain a large portion of a vincular pattern. Section 3.2 explains how the discovery of reversibly deletable sets requires two tests rather than one as well as the introduction of a "null" character. The Maple implementation in GVATTER,[7] provides a practical tool to compute many terms of $s_n(B)$ if B admits a finite scheme of reasonable depth.

Despite the added complications, Theorem 15 proves that any pattern set containing only consecutive patterns and patterns of the form $\sigma_1\sigma_2\cdots\sigma_t\text{-}\sigma_{t+1}$ admits a finite scheme. Hence enumeration schemes may be added to the list of methods to analyze problems in consecutive pattern avoidance. Classical patterns admitting a finite scheme have not been classified, and there have been few results about infinite classes of patterns which admit finite schemes.

In Question 9.2 of [21], Vatter asks "Is every sequence produced by a finite enumeration scheme holonomic (i.e., *P*-recursive)?" It is noted above that 23-1 has a finite scheme which produces the Bell numbers, and Sagan shows in [19] that

---

[6] This refinement is implemented in the Maple package GVATTER as qMiklos.

[7] Available for download from both authors' homepages.

the Bell numbers are not holonomic. Hence a finite scheme can produce a sequence which is not holonomic, although it remains to be seen whether a finite scheme resulting from *classical* patterns (the original context for Vatter's schemes) can yield a non-holonomic sequence. It should be noted that there is no set B of classical patterns for which $s_n(B)$ is known to be non-holonomic, even when B has no known finite enumeration scheme.

Enumeration schemes provide powerful tools for generating terms of the sequence $s_n(B)$ for a broad class of sets B. Thus far they have been developed for vincular patterns and barred patterns. This project originated as an attempt to develop enumeration schemes for the bivincular patterns introduced in [6], but it quickly became apparent that the maps d_R wreak havoc on vertical adjacencies among letters following the prefix and would be unsuitable.[8] A different recursive structure for $\mathcal{S}_n$ would need to be exploited to make enumeration schemes work for bivincular patterns which have non-trivial adjacency requirements.

One may wonder whether these methods may be extended to compute the number of permutations of length n admitting r copies of a given pattern, as done in [14]. The problem considered in this paper would be the $r = 0$ case. Since a given letter can be involved in multiple copies of a pattern, the existence of bijective deletion maps d_R becomes far less likely for short prefixes however. Thus it seems unlikely that the methods contained here will extend easily to the multiple-copies case.

In [16] the second author extends enumeration schemes to pattern avoidance by words, i.e., permutations of multisets. The techniques above should extend in a straightforward manner to this case. Similarly, schemes could be developed to handle permutations (or even words) avoiding barred vincular patterns by combining the techniques of this paper and [18]. These extensions have yet to be implemented by computer.

References

[1] OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences, 2011. http://oeis.org/A095149.
[2] Eric Babson, Einar Steingrímsson, Generalized permutation patterns and a classification of the Mahonian statistics, Sém. Lothar. Combin. 44 (2000) 18 pp.. Art. B44b (electronic).
[3] Andrew Baxter, Refining enumeration schemes to count according to the inversion number, Pure Math. Appl. 21 (2) (2010) 137–160.
[4] Andrew Baxter, From enumeration schemes to functional equations (in preparation).
[5] Andrew M. Baxter, Shape-Wilf-equivalences and vincular patterns (in preparation), arXiv:1201.4767.
[6] Mireille Bousquet-Mélou, Anders Claesson, Mark Dukes, Sergey Kitaev, (2+2)-free posets, ascent sequences and pattern avoiding permutations, J. Combin. Theory Ser. A 117 (7) (2010) 884–909.
[7] Anders Claesson, Generalized pattern avoidance, European J. Combin. 22 (7) (2001) 961–971.
[8] A. Claesson, V. Jelínek, E. Steingrímsson, Upper bounds for the Stanley-Wilf limit of 1324 and other layered patterns. ArXiv e-prints, November 2011.
[9] Sergi Elizalde, Asymptotic enumeration of permutations avoiding generalized patterns, Adv. in Appl. Math. 36 (2) (2006) 138–155.
[10] Sergi Elizalde, Marc Noy, Consecutive patterns in permutations, Adv. in Appl. Math. 30 (1–2) (2003) 110–125. Formal power series and algebraic combinatorics (Scottsdale, AZ, 2001).
[11] J. Fürlinger, J. Hofbauer, q-Catalan numbers, J. Combin. Theory Ser. A 40 (2) (1985) 248–264.
[12] Sergey Kitaev, Multi-avoidance of generalised patterns, Discrete Math. 260 (1–3) (2003) 89–100.
[13] Sergey Kitaev, Partially ordered generalized patterns, Discrete Math. 298 (1–3) (2005) 212–229.
[14] John Noonan, The number of permutations containing exactly one increasing subsequence of length three, Discrete Math. 152 (1–3) (1996) 307–313.
[15] OEIS Foundation Inc. The On-Line Encyclopedia of Integer Sequences, 2011. http://oeis.org.
[16] Lara Pudwell, Enumeration schemes for words avoiding permutations, in: Permutation Patterns, in: London Math. Soc. Lecture Note Ser., vol. 376, Cambridge Univ. Press, Cambridge, 2010, pp. 193–211.
[17] Lara Pudwell, Enumeration schemes for words avoiding patterns with repeated letters, Integers 8 (A40) (2008) 19.
[18] Lara Pudwell, Enumeration schemes for permutations avoiding barred patterns, Electron. J. Combin. 17 (1) (2010) 27. Research Paper 29.
[19] Bruce E. Sagan, Pattern avoidance in set partitions, Ars Combin. 94 (2010) 79–96.
[20] Einar Steingrímsson, Generalized permutation patterns — a short survey, in: Steve Linton, Nik Ruškuc, Vincent Vatter (Eds.), Permutation Patterns, St Andrews 2007, in: London Mathematical Society Lecture Note Series, vol. 376, Cambridge University Press, 2010, pp. 193–211.
[21] Vincent Vatter, Enumeration schemes for restricted permutations, Combin. Probab. Comput. 17 (1) (2008) 137–159.
[22] William Y.C. Chen, Lewis H. Liu, Permutation tableaux and the dashed permutation pattern 32–1, Electron. J. Combin. 18 (1) (2011) Paper 111 11.
[23] Doron Zeilberger, Enumeration schemes and, more importantly, their automatic generation, Ann. Comb. 2 (2) (1998) 185–195.
[24] Doron Zeilberger, On Vince Vatter's brilliant extension of Doron Zeilberger's enumeration schemes for Herb Wilf's classes, The Personal J. Ekhad and Zeilberger (2006).

---

[8] Deleting a letter at the start of the permutation can create a vertical adjacency at the end of the permutation. For example, to contain the bivincular pattern $\sigma = (12, \{1\}, \{1\})$, a permutation π must have a subsequence $\pi_i\pi_{i+1}$ such that $\pi_i + 1 = \pi_{i+1}$. The permutation 13524 avoids σ while $d_{\{2\}}(13524) = 1423$ contains σ in the last two letters.