# Enumeration Schemes for Words Avoiding Patterns with Repeated Letters

Lara Pudwell

November 27, 2007

**Abstract**

In this paper, we discuss the enumeration of words avoiding patterns with repeated letters. More specifically, we find recurrences (i.e. enumeration schemes) counting words avoiding any pattern of length 3 and words avoiding any monotone pattern.

## 1 Background

The enumeration of permutation classes has been accomplished by many beautiful techniques. One natural extension of permutation classes is pattern-avoiding words. In [5], we adapted the method of enumeration schemes, first introduced for permutations by Zeilberger [7] and extended by Vatter [6] to the case of enumerating words avoiding a permutation pattern. In this paper, we modify the enumeration scheme paradigm further to enumerate words avoiding patterns with repeated letters.

First, we recall the following definitions:

**Definition 1.** *Let $w \in [k]^n, w = w_1 \cdots w_n$. The* reduction *of $w$, denoted $red(w)$, is the unique word of length $n$ obtained by replacing the $i^{th}$ smallest entries of $w$ with $i$, for each $i$.*

**Definition 2.** *Let $w \in [k]^n, w = w_1 \cdots w_n$ as above, and let $q \in [k]^m, q = q_1 \cdots q_m$. We say that $w$* contains *$q$ if there exist $1 \leq i_1 < i_2 < \cdots < i_m \leq n$ so that $red(w_{i_1} \cdots w_{i_m}) = q$. Otherwise $w$* avoids *$q$.*

For example, the reduction of $w = 2432 \in [4]^3$ is 1321. Also $w$ contains the pattern 121 as evidenced by the substring $w_1w_2w_4 = 242$, which reduces to 121.

While it is straightforward to fix a word $w$ and list all the patterns $q$ which it contains, it is a more difficult task to fix $q$ and then enumerate the number of words $w$ which do not contain $q$. To this end, we introduce our main object of study:

**Definition 3.** *A* frequency vector *is a vector $\boldsymbol{a} = [a_1, \ldots, a_k]$ such that $k \geq 1$ and $a_i \geq 0$ for $1 \leq i \leq k$. Denote $\|\boldsymbol{a}\| = \sum_{i=1}^{k} a_i$. Then, given a frequency vector $\boldsymbol{a}$ and a set of reduced words $Q$ in $[k]^m$ for some $m > 0$, we define*

$$A(\boldsymbol{a}, Q) := \{w \in [k]^{\|\boldsymbol{a}\|} \mid w \text{ avoids } q \text{ for every } q \in Q, w \text{ has } a_i \ i\text{'s for } 1 \leq i \leq k\}.$$

Notice that if $a_1 = \cdots = a_k = 1$, we reduce to the classical case of counting pattern-avoiding permutations. Further, observe that if $a_i$=0 for some $i$, then $A([a_1, \ldots a_i, \ldots, a_k], Q) = A([a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_k], Q)$, so in general, we may assume that $\mathbf{a}$ has positive entries. When the set of patterns $Q$ is clear from context, we may simply write $A(\mathbf{a})$.

Much of the work studying pattern avoidance in permutations and in words adapts techniques that are dependent on what patterns are being avoided. However, ideally, we want to find a single technique that enumerates many different classes of pattern-avoiding words. In 1998, Zeilberger [7] introduced prefix enumeration schemes to count pattern-avoiding permutations, giving a more universal framework for counting these permutation classes. In 2005, Vatter [6] extended these schemes, completely automating the enumeration of many more permutation classes. Vatter's work studies a symmetry of prefix schemes to ease notation.

The method of Zeilberger's prefix schemes has already been extended to pattern-avoiding words [5] with a reasonable success rate. However, these prefix schemes for words fail completely if one wishes to enumerate words avoiding patterns with repeated letters. First, I will review the limitations of Zeilberger and Vatter's enumeration schemes when they are extended to the case of pattern-avoiding words. Then, I will introduce a new notion of schemes for words. I will use this new notion to find recurrences counting words avoiding *any* pattern of length 3. The main result is that this new notion of scheme is guaranteed to work when enumerating words avoiding *any* monotone pattern of arbitrary length.

## 2 Old Schemes for Permutations

In [5], I extended Zeilberger's notion of prefix scheme to pattern avoiding words.

Suppose we would like to enumerate a set $A(n)$. If we cannot find a closed-form formula for $A(n)$, ideally, we could find a recurrence which depends only on $n$. However, this is not always possible. Following Zeilberger, we introduce the notion of refinement. Namely, parameterize $A(n) = \bigcup_{i \in I} B(n,i)$ for some parameter $i$, so that $A(n)$ is a disjoint union of the $B(n,i)$'s. If we can then find a recurrence for each $B(n,i)$ in terms of the $A(n)$'s and the $B(n,i)$'s, we then have a formula for $A(n)$. If not, continue by parameterizing each $B(n,i) = \bigcup_{j \in J} C(n,i,j)$.

Zeilberger's schemes for pattern-avoiding permutations refine by looking at prefixes. That is if $A(n)$ is the set of words whose first $l$ letters reduce to $p$, then $B(n,i)$ is the set of words whose first $l+1$ letters reduce to some longer prefix $\hat{p} = p_1 \cdots p_l \cdot i$ and such that $red(p_1 \cdots p_l) = p$.

For example, given prefix $p$, let $A_p(\mathbf{a})$ be the set of words with alphabet vector $\mathbf{a} = [a_1, \ldots, a_k]$ whose first $|p|$ letters reduce to $p$. We have:

$$A_\emptyset(\mathbf{a}) = A_1(\mathbf{a}) = A_{12}(\mathbf{a}) \cup A_{11}(\mathbf{a}) \cup A_{21}(\mathbf{a}) = \cdots$$

Furthermore, we can deduce recurrences by knowing the prefix of a word. For example, if we want to avoid the pattern 123, and a 123-avoiding word has prefix 21, the 2 can be deleted, because any possible way for the 2 to be involved in a bad pattern implies that the 1 is also in a bad pattern. Therefore, if the role of 2 is played by a letter $j$, we have $A_{21}([a_1, \ldots, a_j, \ldots, a_k]) = A_1([a_1, \ldots, a_j - 1, \ldots, a_k])$.

While this method of refining based on prefixes and finding recurrences has a reasonable success rate for words avoiding permutations, it was shown in [5] that it necessarily fails if the pattern to be avoided has a repeated letter. Thus, we turn to a symmetric approach introduced by Vatter [6].

## 3 Old Schemes for Permutations: a Symmetry

Vatter's schemes for permutations take a symmetry of this approach and look at the patterns formed by the $i$ smallest letters in a permutation instead of the initial $i$ letters.

In his case, the notation $A_p(\mathbf{a})$ denotes the set of words with alphabet vector $\mathbf{a} = [a_1, \ldots, a_k]$ whose *smallest* $|p|$ letters form pattern $p$. For example, $A_{132}(\mathbf{a})$ denotes the set of words where 1 appears before 3, which appears before 2. Still, we have:

$$A_\emptyset(\mathbf{a}) = A_1(\mathbf{a}) = A_{12}(\mathbf{a}) \cup A_{11}(\mathbf{a}) \cup A_{21}(\mathbf{a}) = \cdots$$

The logic for finding recurrences is similar. For example, if we wish to avoid the pattern 123, and consider the set $A_{21}([a_1, \ldots, a_k])$, we know that if 1 is involved in a 123 pattern, then 2 must also be involved in a 123 pattern, so if the role of 1 is played by $j$, then we have $A_{21}([a_1, \ldots, a_j, \ldots, a_k]) = A_1([a_1, \ldots, a_j - 1, \ldots, a_k]) = A_1([a_1, \ldots a_{j-1}, a_{j+1}, \ldots, a_k])$, where the last equality is because Vatter considers the special case of *permutations*.

In essence, Vatter takes inverses of the permutations in Zeilberger's schemes, and as the inverse map provides an involution on the set of all permutations, this is an equivalent construction.

For words, however, the inverse map no longer exists. Enumerating words by considering the pattern formed by the $i$ smallest letters is no longer as straightforward. In general, to count pattern-avoiding words, the chain of prefixes of smallest letters $1 \to 11 \to 111 \to \cdots$ forms an infinite chain of subsets of $A([a_1, \ldots, a_k])$ without recurrences.

Clearly, while both Zeilberger's and Vatter's schemes are effective for enumerating permutations, they have their drawbacks when extended to words. Thus, we must make more significant modifications.

# 4 New Schemes

Vatter's approach can be modified in the following way. Instead of looking at the patterns formed by the $i$ smallest letters in a word by adding *one* letter at a time, refine by successively adding *all* copies of the smallest letter at once. As we will see, this introduces new parameters into the enumeration scheme, but it allows the enumeration of classes of words which were unable to be enumerated by previous methods.

For example, let $A_\emptyset(\mathbf{a})$ be the set of words with alphabet vector $\mathbf{a} = [a_1, \ldots, a_k]$ avoiding 112. Then, we can refine $A_\emptyset(\mathbf{a}) = A_{11}(\mathbf{a}) \cup A_1(\mathbf{a})$, where $A_{11}(\mathbf{a}) = \{$ the set of words with at least two 1s $\}$, and $A_1(\mathbf{a}) = \{$the set of words with only one 1$\}$. That is, $A_{11}(\mathbf{a})$ is the set of words with enough 1s to be the start of a forbidden 112 pattern, and $A_1(\mathbf{a})$ is the set of words without enough 1s to start a forbidden 112 pattern.

Essentially, instead of tracking the patterns formed by the initial letters of the word (as in Zeilberger's method) or the patterns formed by the smallest letters of the word (as in Vatter's method), we begin with the empty word, and successively insert all copies of a letter at once, and keeping track of the maximal possible subpattern of a forbidden 112 pattern. More explicitly, begin with an empty word and insert all $a_1$ 1s. Keep track of the earliest 11 pattern and insert all $a_2$ 2s. Keep track of the new first 11 pattern, and insert all $a_3$ 3s. Repeat this process until all $a_1 + \cdots + a_k$ letters have been inserted into the word.

Here, we introduce a revised notion of Scheme.

**Definition 4.** *Let $S$ be a set of triples $[A_i, C_i, R_i]$ where*

- *$A_i$ is a set, possibly with extra parameters distinguishing elements of $A_i$.*
- *$C_i$ is a set of pairs $[P_{i,1}, P_{i,2}]$ where each $P_{i,1}$ is a set of $A_j$'s with $j \geq i$, and the $P_{i,2}$'s are disjoint conditions on the parameters of $A_i$.*
- *$R_i$ is a linear combination of sets $A_j, j \leq i$, possibly with coefficients depending on the parameters of $A_i$.*

*We say that $S$ is an* enumeration scheme *if for each triple $[A_i, C_i, R_i]$ in $S$, exactly one of $C_i$ or $R_i$ is non-empty.*

Notice that a scheme can be considered to be an encoding for a system of recurrences. Namely, $C_i$ are the children of $A_i$, so $|A_i| = \sum_{c \in C_i} |c|$, and $R_i$ is a recurrence for $A_i$ in terms of earlier sets, so $|A_i| = R_i$.

A simple example is the following:

$$\{[A_\emptyset([a_1, \ldots, a_k]), \{[A_1([a_1, \ldots, a_k]), (a_1 = 1)], [A_{11}([a_1, \ldots, a_k]), (a_1 > 1)]\}, \emptyset],$$
$$\left[A_1([a_1, \ldots, a_k]), \emptyset, \binom{a_2 + \cdots + a_k + 1}{1} \cdot A_\emptyset([a_2, \ldots, a_k])\right],$$
$$[A_{11}([a_1, \ldots, a_k]), \emptyset, 0]\}$$

This scheme can be interpreted in the following way: Let $A_\emptyset([a_1, \ldots, a_k])$ be the set of all words avoiding 11, $A_1([a_1, \ldots, a_k])$ the set of all words avoiding 11 with $a_1 = 1$, and $A_{11}([a_1, \ldots, a_k])$ the set of all words avoiding 11 where $a_1 > 1$.

If $a_1 = \cdots = a_k = 1$, we have

$$A_\emptyset([a_1, \ldots, a_k]) = A_1([a_1, \ldots, a_k]) = \binom{a_2 + \cdots + a_k + 1}{1} A_\emptyset([a_2, \ldots, a_k])$$
$$= \binom{a_2 + \cdots + a_k + 1}{1} \cdot A_1([a_2, \ldots, a_k]) = \binom{a_2 + \cdots + a_k + 1}{1} \cdot \binom{a_3 + \cdots + a_k + 1}{1} A_\emptyset([a_3, \ldots, a_k])$$
$$= \cdots = \binom{a_2 + \cdots + a_k + 1}{1} \cdot \binom{a_3 + \cdots + a_k + 1}{1} \cdots \binom{a_{k-1} + a_k + 1}{1} \cdot \binom{a_k + 1}{1} = k!.$$

Otherwise, let $j$ be the smallest integer for which $a_j > 1$, and let $(k)_j = k \cdot (k-1) \cdots (k-j+1)$. We have:

$$A_\emptyset([a_1, \ldots, a_j, \ldots, a_k]) = \cdots = (k)_{j-1} \cdot A_\emptyset([a_j, \ldots, a_k]) = (k)_{j-1} \cdot A_{11}([a_j, \ldots, a_k]) = (k)_{j-1} \cdot 0 = 0$$

both as expected.

# 5 Finding Schemes

In Vatter and Zeilberger's schemes, we found recurrences by looking for letters that were *reversibly deletable*, that is, letters that could be deleted from and reinserted into a word without causing a bad pattern. Now we make the notion of *reversibly deletable* more general.

**Definition 5.** *Let $w \in [k]^n$ be an arbitrary word and $p \in [k]^m$ a forbidden pattern written in reduced form. Let $s_i(p)$ denote the substring of $p$ formed by the letters $\leq i$, and set $s_0(p) = \epsilon$, the empty pattern. We say that $w$ is $i$*-critical with respect to *$p$ ($i \geq 0$) if $w$ contains a copy of $s_i(p)$ but avoids $s_{i+1}(p)$.*

For example, let $w = 1431532231$, and let $p = 12324$. Then $s_1(p) = 1$, $s_2(p) = 122$, $s_3(p) = 1232$, and $s_4(p) = 12324$. $w$ is 3-critical, since it contains $s_1(p)$, $s_2(p)$, and $s_3(p)$ as patterns while it avoids the pattern 12324.

Now, we have a more formal way to produce a scheme in the sense of Section 4. Given a forbidden pattern $p \in [k]^m$,

- The $A_i$s are the sets of words that are $i$-critical for $0 \le i \le k-1$, plus $A_k$ (the set of words containing $p$) and $A_\emptyset$ (the set of ALL words avoiding $p$). $A_i$ may include parameters to track the location of a copy of $s_i(p)$.
- If $A_i$ is the set of $i$-critical words, then $C_i$ consists of the pairs $[A_i, ($ conditions to insert new letters while keeping an $i$-critical word $i$-critical $)]$ and $[A_{i+1}, ($ conditions to insert new letters so that an $i$-critical word becomes $(i+1)$-critical $)]$
- $R_i$ results from a case by case analysis of the structure of $i$-critical words. Namely, if there are letters in an $i$-critical word that cannot possibly be involved in a forbidden pattern, they may be deleted. Also, the parameters of $A_i$ that keep track of the location of a copy of $s_i(p)$ within a given word may be adjusted.

As in the case of permutations, the operations of complement and reversal are involutions on the set of words in $[k]^n$ with some useful properties. Namely, if $p$ is a forbidden pattern, $p^c$ is its complement (formed by replacing $i \to k+1-i$), and $p^r$ is its reversal, in the notation of section 1, we have:

$$A_{[a_1,\ldots,a_k]}(\{p\}) = A_{[a_k,\ldots,a_1]}(\{p^c\})$$

$$A_{[a_1,\ldots,a_k]}(\{p\}) = A_{[a_1,\ldots,a_k]}(\{p^r\})$$

Let $Av(p)$ denote the set of all words avoiding $p$. If, we can find a scheme for $Av(p)$, then we have a system of recurrences for counting $Av(p)$, $Av(p^c)$, $Av(p^r)$, and $Av(p^{rc})$.

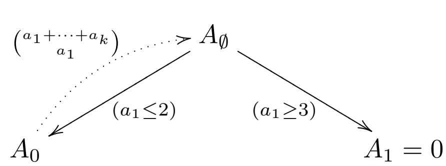
In the following sections, we illustrate the power of this method by finding recurrences to count $Av(p)$, where $p$ is any pattern of length 3.

## 6 Avoiding the pattern 111

The simplest pattern of length 3 is 111. The scheme for $Av(111)$ is very similar to the scheme for $Av(11)$ given in Section 4. Notice that $s_1(111) = 111$, so a word is 1-critical if it contains 3 copies of the same letter.

Let $w \in A_\emptyset([a_1,\ldots,a_k])$ be an arbitrary word in $[k]^{||a||}$. Either $a_1 \le 2$, i.e. $w$ is 0-critical, or $a_1 \ge 3$, in which case $w$ is 1-critical. If a word is 0-critical, the letters in it cannot possibly be part of a bad pattern after subsequent insertion of larger letters, so they may be inserted anywhere in the word, i.e. $|A_1([a_1,\ldots,a_k])| = \binom{a_1+\cdots+a_k}{a_1} \cdot |A_\emptyset([a_2,\ldots,a_k])|$. We represent the situation graphically as follows:

**Figure 1:** $Av(111)$




The nodes in this graph are the sets $A_i$, the solid lines go from $A_i$ to the sets in $C_i$, and are labelled with the corresponding conditions. A labelled dotted arrow contains the information of $R_i$

Or, in the more familiar scheme notation, we have

$$\{[A_\emptyset([a_1,\dots,a_k]),\{[A_0([a_1,\dots,a_k]),(a_1\le 2)],[A_1([a_1,\dots,a_k]),(a_1\ge 3)]\},\emptyset],$$
$$[A_0([a_1,\dots,a_k]),\emptyset,\binom{a_1+\cdots+a_k}{a_1}\cdot A_\emptyset([a_2,\dots,a_k])],$$
$$[A_1([a_1,\dots,a_k]),\emptyset,0]\}$$

We can read this scheme to obtain the following system of recurrences:

If $a_i \le 2$ for all $i$, then

$$A_\emptyset([a_1,\dots,a_k]) = A_0([a_1,\dots,a_k]) = \binom{a_1+\cdots+a_k}{a_1}\cdot A_\emptyset([a_2,\dots,a_k]) = \cdots$$
$$= \binom{a_1+\cdots+a_k}{a_1}\cdots\binom{a_{k-1}+\cdots+a_k}{a_{k-1}}\cdot\binom{a_k}{a_k} = \binom{a_1+\cdots+a_k}{a_1,\dots,a_k} = \{\text{all words with alphabet vector } \mathbf{a}\}$$

Otherwise, let $j$ be the smallest integer for which $a_j \ge 3$. Then

$$A_\emptyset([a_1,\dots,a_k]) = \binom{a_1+\cdots+a_k}{a_1}\cdots\binom{a_{j-1}+\cdots+a_k}{a_{j-1}}\cdot A_\emptyset([a_j,\dots,a_k])$$
$$= \binom{a_1+\cdots+a_k}{a_1}\cdots\binom{a_{j-1}+\cdots+a_k}{a_{j-1}}\cdot A_1([a_j,\dots,a_k]) = \binom{a_1+\cdots+a_k}{a_1}\cdots\binom{a_{j-1}+\cdots+a_k}{a_{j-1}}\cdot 0 = 0$$

# 7 Avoiding the pattern 112

Now, we turn to the case of avoiding patterns of length 3 with 2 distinct letters. Taking into account the symmetries of complement and reversal, once we can count $Av(112)$, we may also count $Av(211)$, $Av(122)$, and $Av(221)$.

Consider the example of 112-avoiding words in more detail. As before, let $A_\emptyset$ be the set of all words avoiding 112, and let $A_0$ and $A_1$ be the sets of 0-critical and 1-critical words respectively. That is, $A_0$ denotes words without a repeated letter, and $A_1$ denotes words with a 11 pattern but no 112 pattern.

We still write $A_\emptyset([a_1,\dots,a_k])$, and $A_0([a_1,\dots,a_k])$ to denote words with a particular alphabet vector, but for $A_1$, we write $A_1([a_i,\dots,a_k],j)$, where $j$ is the position of last letter of the first 11 pattern formed by the letters $1,2,\dots,i-1$ already in the word.

We have the following trivial base cases: If $k=1$, then $A_0([a_1])=1$ (since there is only one word with an alphabet vector $[a_1]$ and it avoids 112), and $A_1([a_i],j)=\binom{j-1+a_i}{a_i}$ (since any letter inserted after the repeated letter in position $j$ forms a 112 forbidden pattern).

Now, consider what happens when $k>1$. In general, we start with the empty word, and insert all $a_1$ copies of 1 into the word. Next, we insert all $a_2$ copies of 2 into the word. At each iteration, the word composed of all letters $1,2,\dots,i$ is called the "old word", and the word composed of $1,2,\dots,i,i+1$, immediately after all copies of the letter $i+1$ have been inserted is called the "new word".

If $k>1$ and $a_1=1$, there is no way for a single smallest letter to be part of a 112 pattern, so we may find the number of words with alphabet vector $[a_2,\dots,a_k]$, and insert the smallest letter anywhere. Thus, $A_0([a_1,\dots,a_k])=\binom{a_2+\cdots+a_k+1}{1}\cdot A_\emptyset([a_2,\dots,a_k])$.

If $k > 1$ and $a_1 > 1$, the first repeated letter in a string of identical letters $\underbrace{1 \dots 1}_{a_1}$ is in position 2, so we have $A_\emptyset([a_1, \dots a_k]) = A_1([a_2, \dots, a_k], 2)$.

Now, we move on to considering the sets $A_1([a_1, \dots, a_k], j)$.

If $k > 1$ and $a_1 = 1$, we may not insert the new (larger) letter after position $j$. There are $j$ choices for where to insert this letter into the word before position $j$. Moreover, inserting this letter in the beginning of the word moves the first repeated letter to position $j+1$ (see figures 2 and 3). Thus, we have $A_1([a_1, \dots, a_k], j) = j \cdot A_1([a_2, \dots, a_k], j + 1)$.

**Figure 1: Old word, before inserting $a_1$ 2s**

| | | |
|---|---|---|
| | $j-1$ old letters | 1 |
| *position* | | $j$ |

**Figure 2: New word, after inserting $a_1$ 2s**

| | | |
|---|---|---|
| | $j-1$ old letters<br>$+1$ new letter | 1 |
| *position* | | $j+1$ |

If $k > 1$ and $a_1 > 1$, again we know that none of the $a_1$ (larger) letters to be inserted may appear after position $j$. Since there are at least two identical letters to insert before position $j$, the new first repeated letter will be one of the newly inserted letters. Thus, let the new first repeated letter be in position $l$ (as in figure 5). There are $l - 2$ old letters and 1 new letter before position $l$, and there are $(j - 1) - (l - 2)$ old letters and $a_1 - 2$ old letters between position $l$ and the old first repeated letter in position $j$. Thus, summing over all posibilities for position $l$, $A_1([a_1, \dots, a_k], j) = \sum_{l=2}^{j+1} (l-1) \binom{(i-1)+(l-2)+(a_1-2)}{(a_1-2)} A_1([a_2, \dots, a_k], j)$.

**Figure 4: Old word, before inserting $a_1$ 2s**

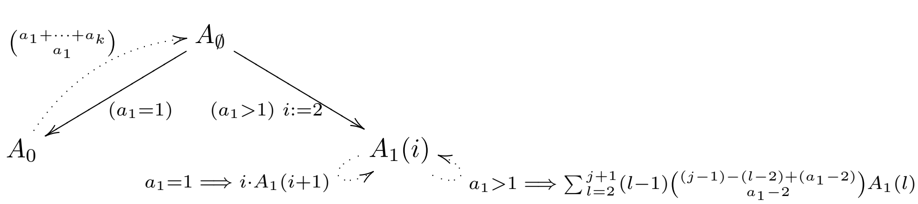
| | | |
|---|---|---|
| | $(j-1)$ old letters | 1 |
| *position* | | $j$ |

**Figure 5: New word, after inserting $a_1$ 2s**

| | | | | |
|---|---|---|---|---|
| | $l-2$ old letters<br>$+1$ new letter | 2 | $(j-1)-(l-2)$ old letters<br>$+(a_1-2)$ new letters | 1 |
| *position* | | $l$ | | $j+a_1$ |

Graphically, we have

**Figure 6: Av(112)**



Which is the same as:

$$A([a_1,\ldots,a_k]) = \begin{cases} 1 & k=1 \\ \binom{a_2+\cdots+a_k+1}{1} A([a_2,\ldots,a_k]) & k>1, a_1=1 \\ B([a_2,\ldots,a_k],2) & k>1, a_1>1 \end{cases}$$

$$B([a_1,\ldots,a_k],j) = \begin{cases} \binom{j-1+a_1}{a_1} & k=1 \\ j * B([a_2,\ldots,a_k],j+1) & a_1=1 \\ \sum_{l=2}^{j+1}(l-1)\binom{(j-1)-(l-2)+(a_1-2)}{a_1-2} B([a_2,\ldots,a_k],l) & a_1>1 \end{cases}$$

This recurrence is uniquely satisfied by

$$A_{[a_1,\ldots,a_k]} = \prod_{i=2}^{k}(a_i+\cdots+a_k+1)$$

This is a new proof of a result given by Heubach and Mansour [4]. More significantly it can be easily generalized, as we will see.

# 8 Avoiding the pattern 121

To completely count all patterns of length 3 with at most 2 letters, it remains to count $Av(121)$ (which will allow us to enumerate $Av(212)$).

We can do this easily by adding a new parameter. The algorithm remains the same. Begin with an empty word, and insert all copies of the smallest letter. Then, consider how many ways to insert the next largest letter, keeping track of the maximal bad pattern. Since 121 is not a monotone pattern, however, it no longer suffices to keep track of the earliest 11 pattern in 1-critical words.

More specificially, when we consider all copies of a letter $l$ in a 121-avoiding word, we know that there can be no larger letters between the first $l$ and the last $l$ in the word. Thus, this first $l$, last $l$, and all letters in between act as if they were only one letter. Instead of parameterizing our scheme in terms of locations of letters, it suffices to keep track of the number of such "blocks" of letters in the word. Notice that a word may be either 0-critical or 1-critical from our previous notation and still have any number of blocks. Thus, for 1-critical words, we define the following:

$$A_1(\mathbf{a},i) := \{\text{1-critical words with alphabet vector } \mathbf{a} \text{ and exactly } i \text{ blocks of letters }\}$$

so $A_\emptyset([a_1,\ldots,a_k]) := \{\textit{all } \text{121-avoiding words}\} = A_1([a_2,\ldots,a_k],1)$. i.e., a word consisting of $a_1$ 1s consists of a single block.

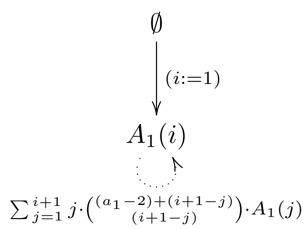
Now, consider a word with $i$ blocks. We may not insert new letters into the middle of a block, but we may insert letters anywhere between blocks. Moreover, if the new letters are not all adjacent, the first new letter and the last new letter form the beginning and end of a new block.

For example, suppose the current 121-avoiding word is 33211222 and we wish to insert 2 4s. The current word has 2 blocks: 33, and 211222. So we may put the 4s together: 4433211222, 3344211222 or 3321122244, or we may separate them 4334211222, 4332112224, 3342112224.

In general, suppose we have a word $w \in A([a_1, \dots, a_{l-1}])$ with $i$ blocks, and we wish to insert $a_l$ copies of the letter $l$. The position of the first new $l$ and the last new $l$ determine a new block. Suppose that between these two new letters there were $b$ old blocks. Then there are $\binom{(a_l-2)+(b)}{b}$ ways to arrange the other new letters inside this new block. Moreover this turned $b$ blocks into 1 block for a net loss of $b-1$ blocks. So if $j$ is the new number of blocks after letter insertions, there are $j = i-(b-1)$, i.e. $b = i+1-j$. Now there are $j$ ways to pick which consecutive $b$ blocks will become one single new block, so there are $j \cdot \binom{(a_l-2)+(i+1-j)}{(i+1-j)}$ ways to get $j$ blocks from $i$ blocks by inserting $a_l$ letters.

This is represented graphically in figure 7, and can be written as:

**Figure 7: Av(121)**




$$A_1(i)[a_1, \dots, a_k] = \sum_{j=1}^{i+1} j \cdot \binom{(a_1-2)+(i+1-j)}{(i+1-j)} A_1(j)[a_2, \dots, a_k]$$

Together with the base case of $A_1(i)[] = 1$, for the empty word, we have a recurrence completely counting all words avoiding the pattern 121 which yields the unique solution:

$$A_{\emptyset}[a_1, \dots, a_k] = \prod_{i=2}^{k}(a_i + \cdots + a_k + 1)$$

This result was also given by Heubach and Mansour [4], but was shown in a different way.

# 9 Avoiding the pattern 213

We now move on to words avoiding patterns of length 3 with 3 letters, i.e. words avoiding permutations. This case can be taken care of by prefix schemes, as noted in [5], but for the sake of completeness, we describe an alternate enumeration using the method of this paper.

Again, by the symmetries of complement and reversal, an enumeration scheme for $Av(213)$ allows us to count $Av(312)$, $Av(132)$, and $Av(231)$.

We return to our original notation of $i$-critical words, and add a few parameters.

Let:

$$A_{\emptyset}([a_1, \dots, a_k]) = \{\text{all words}\}$$

$$A_1([a_1, \dots, a_k], p) = \{\text{ all 1-critical words with } p \text{ letters after the left-most 1 pattern}\}$$

$$A_2([a_1, \dots, a_k], p) = \{\text{ all 2-critical words with leftmost 21 pattern ending in position } p\}$$

Trivially, by inserting $a_1$ identical letters into the empty word, we have $A_\emptyset([a_1,\dots,a_k]) = A_1([a_2,\dots,a_k],a_1-1)$.

Now consider an arbitrary 1-critical word. Since this word contains a 1 pattern, but not a 21 pattern, all letters must be in increasing order.

Consider a generic member of $A_1([a_1,\dots,a_k],p)$, as in figure 8. When we insert $a_1$ new letters into this word, either (a) we do not create a new 21 pattern (i.e. all new letters are appended to the end of the word), as in figure 9, or (b), we do create a new 21 pattern, and keep track of where the leftmost such pattern ends, as in figure 10.

**Figure 8: Generic member of $A_1([a_1,\dots,a_k],p)$**

| | 1 | $p$ old letters |
|---|---|---|
| *position* | 1 | |

**Figure 9: Case a: no new 21 pattern**

| | 1 (old letter) | $p$ old letters | $a_1$ new letters |
|---|---|---|---|
| *position* | 1 | | |

**Figure 10: Case b: new 21 pattern induced**

| | $j$ old letters | 2 (new letter) | $l-1$ new letters | 1 (old letter) | $p-j$ old letters $+a_1-l$ new letters |
|---|---|---|---|---|---|
| *position* | | $j+1$ | | $j+l+1$ | |

Thus, $A_1([a_1,\dots,a_k],p) = A_1([a_2,\dots,a_k],p+a_1) + \sum_{l=1}^{a_1}\sum_{j=0}^{p}\binom{(p-j)+(a_1-l)}{a_1-l}\cdot A_2([a_2,\dots,a_k],j+l+1)$.

Finally, consider all 2-critical words, that is, words that contain a 21 pattern, but not a 213.

Say that the leftmost 21 ends in position $p$. Then no new (larger) letters may be inserted after position $p$ without creating a forbidden 213 pattern. Again, either (a) the letter that plays the role of 1 in the current leftmost 21 pattern stays the same, as in figure 11, or (b) the newly inserted letters create an earlier 21 pattern, as in figure 12.

**Figure 11: case a: 2-critical word with same leftmost 21 pattern**

| | $p-1$ old letters | $a_1$ new letters | 1 |
|---|---|---|---|
| *position* | | | $p+a_1$ |

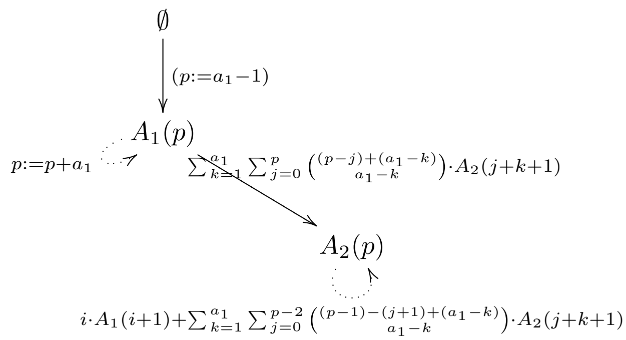
**Figure 12: case b: 2-critical word with new leftmost 21 pattern**

| | $j$ old letters | 2 (new letter) | $l-1$ new letters | 1 (old letter) | $(p-1)-(j+1)$ old letters $+a_1-l$ new letters | old 1 |
|---|---|---|---|---|---|---|
| *position* | | $j+1$ | | $j+l+1$ | | |

Thus, $A_2([a_1,\dots,a_k],p) = A_2([a_2,\dots,a_k],p+a_1)+\sum_{l=1}^{a_1}\sum_{j=0}^{p-2}\binom{(p-1)-(j+1)+(a_1-l)}{a_1-l}\cdot A_2([a_2,\dots,a_k],j+l+1)$.

We can represent this in the more compact graphical notation, as in figure 13.

**Figure 13: Av(213)**




Although this does not readily yield a nice closed formula as in the previous examples, we have now deduced a system of recurrences that completely enumerates all words avoiding 213. This is an alternative way to enumerate these words from Burstein [2].

## 10 Avoiding the pattern 123

To complete our classification of words avoiding patterns of length 3, we examine words avoiding the permutation 123. The analysis of $Av(123)$ turns out to be very similar to the analysis of $Av(213)$.

Let:

$$A_\emptyset([a_1,\ldots,a_k]) = \{\text{all words}\}$$
$$A_1([a_1,\ldots,a_k],p) = \{\text{ all 1-critical words with } p \text{ letters after the left-most 1 pattern}\}$$
$$A_2([a_1,\ldots,a_k],p) = \{\text{ all 2-critical words with leftmost 12 pattern ending in position } p\}$$

Trivially, by inserting $a_1$ identical letters into the empty word, we have $A_\emptyset([a_1,\ldots,a_k]) = A_1([a_2,\ldots,a_k],a_1-1)$.

Now consider an arbitrary 1-critical word. Since this word contains a 1 pattern, but not a 12 pattern, all letters must be in decreasing order, as in figure 14. Notice, that we keep the leftmost 1 pattern separate for further analysis.

When we insert $a_1$ new letters into this word, either (a) we do not create a new 12 pattern (i.e. all new letters are appended to the beginning of the word), as in figure 15, or, (b), we do create a new 12 pattern, and keep track of where leftmost such pattern ends, as in figure 16.

**Figure 14: Generic member of $A_1([a_1,\ldots,a_k],p)$**

| | 1 | $p$ old letters |
|---|---|---|
| *position* | 1 | |

**Figure 15: Case a: no new 12 pattern**

| | 1 (new letter) | $(a_1-1)$ new letters | old 1 | $p$ old letters |
|---|---|---|---|---|
| *position* | 1 | | $a_1+1$ | |

**Figure 16: Case b: new 12 pattern is induced**

| | | | | | |
|---|---|---|---|---|---|
| | $l-1$ new letters | 1 (old letter) | $j$ old letters | 2 new letter | $p-j$ old letters $+a_1-l$ new letters |
| *position* | | $l$ | | $j+l+1$ | |

Thus, $A_1([a_1,\dots,a_k],p) = A_1([a_2,\dots,a_k],p+a_1) + \sum_{l=1}^{a_1}\sum_{j=0}^{p}\binom{(p-j)+(a_1-l)}{a_1-l}\cdot A_2([a_2,\dots,a_k],j+l+1)$.

Finally, consider all 2-critical words, that is, words that contain a 12 pattern, but not a 123. A generic 2-critical word is shown in figure 17.

Say that the leftmost 12 ends in position $p$. Then no letters may be inserted after position $p$ without creating a forbidden 123 pattern. Again, either (a) the letter that plays the role of 2 in the current leftmost 12 pattern stays the same, as in figure 18, or (b) the newly inserted letters create an earlier 12 pattern, as in figure 19.

**Figure 17: generic 2-critical word**

| | | | |
|---|---|---|---|
| | $p-1$ old letters including "1" | 2 (from leftmost 12 pattern) | remaining old letters |
| *position* | | p | |

**Figure 18: Case a: same leftmost 12 pattern**

| | | | |
|---|---|---|---|
| | $a_1$ new letters | $p-1$ old letters | 2 (old letter) |
| *position* | | | $p+a_1$ |

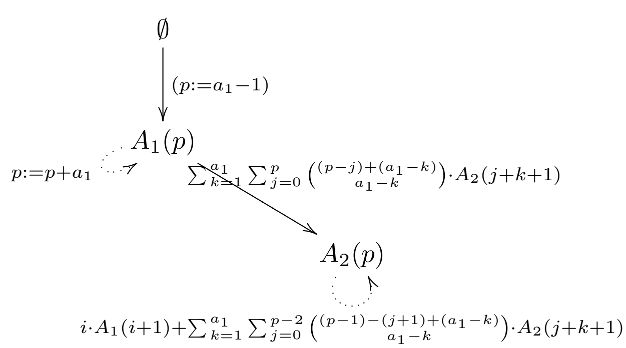
**Figure 19: Case b: earlier leftmost 12 pattern is induced**

| | | | | | | |
|---|---|---|---|---|---|---|
| | $l-1$ new letters | 1 (old letter) | $j$ old letters | 2 (new letter) | $(p-1)-(j+1)$ old letters $+a_1-l$ new letters | old 2 |
| *position* | | $l$ | | $j+l+1$ | | |

Thus, $A_2([a_1,\dots,a_k],p) = A_2([a_2,\dots,a_k],p+a_1) + \sum_{l=1}^{a_1}\sum_{j=0}^{p-2}\binom{(p-1)-(j+1)+(a_1-l)}{a_1-l}\cdot A_2([a_2,\dots,a_k],j+l+1)$.

This can be represented using the more compact graphical notation as in figure 20.

**Figure 20: Av(123)**




The fact that $Av(123)$ and $Av(132)$ are equinumerous, as noted by Burstein [2], can be illustrated in a new way via these identical schemes.

Now that we have used our divide and conquer method of finding schemes for words avoiding any pattern of length 3, we turn to the main theorem of this paper.

## 11 Avoiding Monotone Patterns

The results of this paper, to this point, have previously been shown using other methods. As the case by case analysis involved in finding schemes for pattern-avoiding words seems to be quite tedious at times, one may wonder what the advantage of the present method is.

To date, there had been no infinite family of classes of pattern avoiding words (or permutations) which have been shown to each have a finite enumeration scheme. This new kind of scheme has the advantage that there provably exists a scheme for words avoiding *any* monotone pattern.

For ease of notation, consider the monotone pattern $p = 1^{b_1} \cdots m^{b_m}$. Let $A_\emptyset([a_1, \ldots, a_k])$ be the set of all words avoiding $p$ with frequency vector $[a_1, \ldots, a_k]$. As before, $A_i([a, \ldots, a_k])$ is the set of $i$-critical words with respect to $p$.

If $a_1 < b_1$, then we have $A_\emptyset([a_1, \ldots, a_k]) = \binom{a_2+\cdots+a_k+1}{a_1} A_\emptyset([a_2, \ldots, a_k])$, and also $A_i([]) = 1$ for any $i$.

For $A_i$, we also keep track of the positions of the end of the leftmost $1^{b_1}$, $1^{b_1}2^{b_2}$, $1^{b_1}2^{b_2}3^{b_3}$, …, $1^{b_1}2^{b_2}\cdots i^{b_i}$ patterns.

Now, to find a recurrence equation for each $A_i$, one must only complete a case by case counting exercise, as in the examples above. That is, consider the cases:

- the insertion of new (larger) letters does not affect any of the existing $1^{b_1}$, $1^{b_1}2^{b_2}$, $1^{b_1}2^{b_2}3^{b_3}$, …, $1^{b_1}2^{b_2}\cdots i^{b_i}$ patterns
- the new letters create a new leftmost $1^{b_1}2^{b_2}$ pattern, but do not affect any of the $1^{b_1}2^{b_2}3^{b_3}$, …, $1^{b_1}2^{b_2}\cdots i^{b_i}$ patterns
- the new letters create a new leftmost $1^{b_1}2^{b_2}3^{b_3}$ pattern, but do not affect any of the $1^{b_1}2^{b_2}3^{b_3}4^{b_4}$,…, $1^{b_1}2^{b_2}\cdots i^{b_i}$ patterns
- …
- the new letters create new leftmost $1^{b_1}2^{b_2}$, $1^{b_1}2^{b_2}3^{b_3}$, …, $1^{b_1}2^{b_2}\cdots i^{b_i}$ patterns

Although the counting and notation may get quite hairy, there are no added subtleties: to count words avoiding a monotone pattern, one must only take sums of combinations of old sets as shown above.

**Theorem 1.** *The set of words avoiding the monotone pattern $1^{b_1} \cdots m^{b_m}$ has a scheme consisting of at most $m$ triples $[A_i, C_i, R_i]$.*

*Proof.* For each $A_i$ above, $A_i$ can clearly be written as a combination of $A_i$s (either adding new letters does not create an $(i+1)$-critical pattern) and $A_{i+1}$s (adding new letters creates at worst an $(i+1)$-critical pattern). Since $A_i = 0$ for $i \geq m$, the scheme ends with state $A_{m-1}$, which has no new children (because words in $A_m$ contain pattern $p$) thus giving a scheme of $m$ triples. □

It should be noted that this is the first method that *guarantees* a way to count a non-trivial family of classes of pattern-avoiding words of arbitrary length. It should also be noted that the case of enumerating pattern avoiding permutations avoiding a monotone pattern is a special case of this theorem, given by setting $a_1 = \cdots a_k = b_1 \cdots b_m = 1$.

## 12 Future Work

The new version of enumeration schemes described in this paper gives a way to count words avoiding *any* pattern of length up to 3. Further, the simple structure of monotone patterns makes it easy to track occurences of subpatterns. Thus, words avoiding *any* monotone pattern are *guaranteed* to be counted by this method.

Together with the method of prefix schemes for words given in [5], one can find recurrences counting many classes of pattern-avoiding words. However, the following questions still remain:

- Can this method of schemes for monotone patterns be modified to count words avoiding non-monotone patterns?
- What are other methods to count many classes of pattern-avoiding words?

## References


[1] Petter Branden, Toufik Mansour, *Finite automata and pattern avoidance in words*, Journal Combinatorial Theory Series A 110:1 (2005) 127-145.

[2] Alexander Burstein, *Enumeration of Words with Forbidden Patterns*, Ph.D. Thesis, University of Pennsylvania, 1998.

[3] Alexander Burstein, Toufik Mansour, *Words restricted by patterns with at most 2 distinct letters*, Electron. J. Combin. 9(2) (2002-2003), #R3.

[4] Silvia Heubach, Toufik Mansour, *Avoiding patterns of length three in compositions and multiset permutations*, Advances in Applied Mathematics **36:2** (2006), 156-174 .

[5] Lara Pudwell, *Enumeration Schemes for Words Avoiding Permutations*, To Appear.

[6] Vince Vatter, *Enumeration Schemes for Restricted Permutations*, Combinatorics, Probability, and Computing, to appear.

[7] Doron Zeilberger, *Enumeration Schemes, and More Importantly, Their Automatic Generation*, Annals of Combinatorics **2** (1998), 185-195.

[8] Doron Zeilberger, *On Vince Vatter's Brilliant Extension of Doron Zeilberger's Enumeration Schemes for Herb Wilf's Classes*, The Personal Journal of Ekhad and Zeilberger, 2006. `http://www.math.rutgers.edu/∼ zeilberg/pj.html`.