Increasing Consecutive Patterns in Words

Mingjia Yang
Department of Mathematics
Rutgers University (New Brunswick)
Piscataway, NJ 08854
USA
my237@math.rutgers.edu
Doron Zeilberger
Department of Mathematics
Rutgers University (New Brunswick)
Piscataway, NJ 08854
USA
DoronZeil@gmail.com

Abstract

We show how to enumerate words in $1^{m_1} \cdots n^{m_n}$ that avoid the increasing consecutive pattern $12 \cdots r$ for any $r \geq 2$. Our approach yields an $O(n^{s+1})$ algorithm to enumerate words in $1^s \cdots n^s$, avoiding the consecutive pattern $1 \cdots r$, for *any* s, and *any* r. This enables us to supply many more terms to quite a few OEIS sequences, and create new ones. We also treat the more general case of counting words with a specified number of the pattern of interest (the avoiding case corresponding to zero appearances). This article is accompanied by three Maple packages implementing our algorithms.

1 Introduction

Simion and Wilf initiated the study of enumerating *classical* pattern-avoidance. This is a very dynamic area with its own annual conference.

Recall that a permutation $\pi = \pi_1 \cdots \pi_n$ avoids a pattern $\sigma = \sigma_1 \cdots \sigma_k$ if none of the $\binom{n}{k}$ length-k subsequences of π, reduces to σ.

Burstein [2], in a 1998 PhD thesis, under the direction of Wilf, pioneered the enumeration of *words* avoiding a set of patterns. This field is also fairly active today, with notable contributions by, *inter alia*, Mansour [3] and Pudwell [12].

The enumeration of permutations avoiding a given (classical) pattern, or a set of patterns, is notoriously difficult, and it is widely believed to be intractable for most patterns, hence it would be nice to have other notions for which the enumeration is more feasible. Such an analog was given, in 2003, by Elizalde and Noy, in a seminal paper [5], that introduced the study of the enumeration of permutations avoiding *consecutive* patterns. A permutation $\pi = \pi_1 \cdots \pi_n$ avoids a consecutive pattern $\sigma = \sigma_1 \cdots \sigma_k$ if none of the $n - k + 1$ length-k consecutive subwords, $\pi_i \pi_{i+1} \cdots \pi_{i+k-1}$ of π, reduces to σ.

Algorithmic approaches to the enumeration of *permutations* avoiding sets of consecutive patterns were given by Nakamura, Baxter, and Zeilberger [10, 1]. Our present approach may be viewed as an extension, from permutations to words, of Nakamura's paper, who was also inspired by the Goulden-Jackson cluster method, but in a sense, is more straightforward, and closer in spirit to the original Goulden-Jackson cluster method ([8], that is beautifully exposited (and extended!) in [11]).

In this article we will consider consecutive patterns of the form $1 \cdots r$, i.e. *increasing consecutive patterns*, and show how to count words in $1^{m_1} \cdots n^{m_n}$ avoiding the pattern $1 \cdots r$ (Theorem 1, that is due to Ira Gessel [6]). Throughout this article we will only consider consecutive patterns, so the word "consecutive" may be omitted. In particular, we will look at how to efficiently count words in $1^s \cdots n^s$ avoiding the pattern $1 \cdots r$. All the sequences for $s = 1$ and $3 \leq r \leq 9$ are in the On-Line Encyclopedia of Integer Sequences, with many terms. Also, quite a few of theses sequences for $s > 1$ are already there, but with very few terms. Our implied algorithms are $O(n^{s+1})$ and hence yield many more terms, and, of course, new sequences.

In the last part of the paper, we will provide a new proof of Theorem 1 by tweaking the Goulden-Jackson cluster method. Using this proof, along with a little more effort, we will generalize Theorem 1 to counting words with a specified number of the pattern $12\cdots r$ (Theorem 3), instead of just *avoiding*, that is, having *zero* occurrence of the pattern of interest.

We close this introduction by mentioning the pioneering work of Mendes and Remmel [9], in combining the two keywords "consecutive patterns" and "words". We were greatly inspired by their article, but our focus is algorithmic.

Maple Packages: This article is accompanied by three Maple packages available from the webpage:

`http://sites.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/icpw.html` .

These are

- `ICPW.txt`: For fast enumeration of sequences enumerating words avoiding increasing consecutive patterns.
- `ICPWt.txt`: For fast computation of sequences of weight-enumerators for words according to the number of increasing consecutive patterns ($t = 0$ reduces to the former case).
- `GJpats.txt`: For conjecturing generating functions (that still have to be proved by humans).

This page also has links to numerous input and output files. The input files can be modified to generate more data, if desired.

2 Method, experimentation, and results

2.1 The Goulden-Jackson cluster method

Recall that the original Goulden-Jackson method [8, 11] inputs a *finite* alphabet, A, that may be taken to be $\{1,\dots,n\}$, and a finite set of "bad words", B.

It outputs a certain **rational function**, let us call it $F(x_1,\dots,x_n)$, that is the multi-variable generating function, in $x_1,\dots,x_n$, for the discrete n-variable function

$$f(m_1,\dots,m_n) \quad ,$$

that counts the words in $1^{m_1}\cdots n^{m_n}$ (there are altogether $(m_1+\cdots+m_n)!/(m_1!\cdots m_n!)$ of them) that **never** contain as *consecutive* subwords (aka *factors* in linguistics) any member of B. In other words:

$$F(x_1,\dots,x_n) = \sum_{(m_1,\dots,m_n)\in N^n} f(m_1,\dots,m_n)\, x_1^{m_1}\cdots x_n^{m_n} \quad .$$

This is nicely implemented in the Maple package `DavidIan.txt`, that accompanies [11], and is freely available from

`http://sites.math.rutgers.edu/~zeilberg/tokhniot/DavidIan.txt` .

For example, let $n = 4$, so the alphabet is $\{1, 2, 3, 4\}$, and let the set of "bad words" to avoid be $\{1234, 1432\}$. Starting a Maple session and typing

```
read 'DavidIan.txt':  lprint(subs(t=0,GJgf(1,2,3,4,[1,2,3,4],[1,4,3,2],x,t)));
```

immediately returns

```
1/(1-x[1]-x[2]-x[3]-x[4]+ 2*x[1]*x[2]*x[3]*x[4])
```
,

that in Human language reads

$$\frac{1}{1 - x_1 - x_2 - x_3 - x_4 \, + \, 2\, x_1 x_2 x_3 x_4} \quad .$$

2.2 Enumerating words avoiding consecutive patterns: let the computer do the guessing

Now we are interested in words in an *arbitrarily large* alphabet $\{1, \ldots, n\}$ avoiding a set of consecutive patterns, but each pattern, e.g., 123, entails an *arbitrarily large* set of forbidden consecutive subwords. For example, in this case, the set of forbidden consecutive subwords is

$$\{i_1\, i_2\, i_3 \,|\, 1 \le i_1 < i_2 < i_3 \le n\} \quad .$$

We can ask `DavidIan.txt` to find the generating function for each specific n, and then hope to conjecture a **general** formula in terms of $x_1, \ldots, x_n$, for *general* (i.e., **symbolic**) n.

This is accomplished by the Maple package `GJpats.txt`, available from the webpage of this article. It uses the original `DavidIan.txt` to produce the corresponding generating functions for increasing values for n, and then attempts to conjecture a *meta-pattern*. For example for words avoiding the consecutive pattern 123 (alias the word 123), for $n = 3$,

`GFpats(`$\{[1, 2, 3]\}, x, 3, 0$`);` (the 0 stands for having zero occurrences of (i.e., avoiding) the pattern of interest) yields

$$1/(1 - x_1 - x_2 - x_3 + x_1x_2x_3) \quad .$$

This is simple enough. Moving right along,

`GFpats(`$\{[1, 2, 3]\}, x, 4, 0$`);` yields

$$1/(1 - x_1 - x_2 - x_3 - x_4 + x_1x_2x_3 + x_1x_2x_4 + x_1x_3x_4 + x_2x_3x_4 - x_1x_2x_3x_4) \quad ,$$

while `GFpats(`$\{[1, 2, 3]\}, x, 5, 0$`);` yields

$$1/(1 - x_1 - x_2 - x_3 - x_4 - x_5 + x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_3x_4 + x_1x_3x_5 + x_1x_4x_5+$$

$$x_2x_3x_4+x_2x_3x_5+x_2x_4x_5+x_3x_4x_5-x_1x_2x_3x_4-x_1x_2x_3x_5-x_1x_2x_4x_5-x_1x_3x_4x_5-x_2x_3x_4x_5) \quad .$$

These look like *symmetric* functions. Procedure `SPtoM(P,x,n,m)` expresses a polynomial, P, in the indexed variables $x[1], \ldots, x[n]$ in terms of the *monomial symmetric polynomials* m_λ. Applying this procedure we have

`SPtoM(denom(GFpats(`$\{[1, 2, 3]\}, x, 5, 0)), x, 5, m$`);` yields

```
-m[1, 1, 1, 1] + m[1, 1, 1] - m[1] + m[]   .
```

`SPtoM(denom(GFpats({[1, 2, 3]}, x, 6, 0)), x, 6, m);` yields

```
m[1,1,1,1,1,1]-m[1,1,1,1]+m[1,1,1]-m[1]+m[]   .
```

`SPtoM(denom(GFpats({[1, 2, 3]}, x, 7, 0)), x, 7, m);` yields

```
-m[1,1,1,1,1,1,1]+m[1,1,1,1,1,1]-m[1,1,1,1]+m[1,1,1]-m[1]+m[]   .
```

You do not have to be a Ramanujan to conjecture the following result.

Fact: The generating function for words in $\{1, 2, \ldots, n\}$ avoiding the consecutive pattern 123, let us call it $F_3(x_1, \ldots, x_n)$ is

$$F_3(x_1, \ldots, x_n) = \frac{1}{1 - e_1 + e_3 - e_4 + e_6 - e_7 + e_9 - e_{10} + \cdots} \quad ,$$

where e_i stands for the *elementary symmetric function* of degree i in $x_1, \ldots, x_n$, i.e., the coefficient of z^i in $(1 + x_1 z) \cdots (1 + x_n z)$. (Note that $e_i = m_{1^i}$.)

Doing the analogous guessing for the consecutive patterns 1234 and 12345, a *meta-pattern* emerges, and we were safe in formulating the following theorem that we discovered using the present experimental mathematics approach. After the first version of this article was posted, we found out, thanks to Justin Troyka, that this theorem is due to Ira Gessel [6, p. 51].

Theorem 1. *(Gessel [6]) For $n \geq 1, r \geq 2$, the generating function for words in $\{1, 2, \ldots, n\}$ avoiding the consecutive pattern $12 \cdots r$, let us call it $F_r(x_1, \ldots, x_n)$ is*

$$F_r(x_1, \ldots, x_n) = \frac{1}{1 - e_1 + e_r - e_{r+1} + e_{2r} - e_{2r+1} + e_{3r} - e_{3r+1} + \cdots} \quad .$$

Of course, if Gessel did not prove it before us, these would have been "only" guesses, but once known, humans can prove them. We did it by tweaking the cluster method to apply to an *arbitrarily large* alphabet, i.e. where even the size of the alphabet, n, is *symbolic*. Our proof of Gessel's theorem will be given at the end of this article.

2.3 Efficient computations

Theorem 1 immediately implies the following partial recurrence equation for the actual coefficients.

Fundamental Recurrence: Let $f_r(\mathbf{m})$ be the number of words in the alphabet $\{1, \ldots, n\}$ with m_1 1's, m_2 2's, $\ldots$, m_n n's (where we abbreviate $\mathbf{m} = (m_1, \ldots, m_n)$) that avoid the consecutive pattern $1 \cdots r$. Also let V_i be the set of $0 - 1$ vectors of length n with i ones, then

$$\begin{aligned} f_r(\mathbf{m}) = & \sum_{\mathbf{v} \in V_1} f_r(\mathbf{m} - \mathbf{v}) - \sum_{\mathbf{v} \in V_r} f_r(\mathbf{m} - \mathbf{v}) \\ & + \sum_{\mathbf{v} \in V_{r+1}} f_r(\mathbf{m} - \mathbf{v}) - \sum_{\mathbf{v} \in V_{2r}} f_r(\mathbf{m} - \mathbf{v}) \end{aligned}$$

$$+\sum_{\mathbf{v}\in V_{2r+1}} f_r(\mathbf{m}-\mathbf{v}) - \sum_{\mathbf{v}\in V_{3r}} f_r(\mathbf{m}-\mathbf{v})$$
$$+\sum_{\mathbf{v}\in V_{3r+1}} f_r(\mathbf{m}-\mathbf{v}) - \sum_{\mathbf{v}\in V_{4r}} f_r(\mathbf{m}-\mathbf{v}) + \cdots \quad .$$

(Readers can check this derivation by multiplying each side of the equation in Theorem 1 by the denominator of the right hand side and then using the fact that $F_r(x_1,\ldots,x_n) = \sum_{(m_1,\ldots,m_n)\in N^n} f_r(m_1,\ldots,m_n)\, x_1^{m_1}\cdots x_n^{m_n}$.)

Suppose that we want to compute $f_3(1^{100})$, i.e., the number of permutations of length 100 that avoid the consecutive pattern 123. If we use the above recurrence literally, we would need about 2^{100} computations, but there is a shortcut!

It follows from the symmetry of the generating function $F_r(x_1,\ldots,x_n)$, that $f_r(m_1,\ldots,m_n)$ is symmetric, hence the above Fundamental Recurrence immediately implies the following recurrence, that enables a very fast computation of the sequences, let us call them $a_r(n)$, for the number of *permutations* of length n that avoid the consecutive pattern $1\cdots r$.

2.3.1 Fast recurrence for enumerating permutations avoiding the consecutive pattern $1\cdots r$

$$a_r(n) = na_r(n-1) - \binom{n}{r}a_r(n-r) + \binom{n}{r+1}a_r(n-r-1) - \binom{n}{2r}a_r(n-2r) + \binom{n}{2r+1}a_r(n-2r-1)$$
$$-\binom{n}{3r}a_r(n-3r) + \binom{n}{3r+1}a_r(n-3r-1) - \cdots \quad .$$

This recurrence goes back to F. N. David and D. Barton [4, p. 157], whose proof used a probabilistic language and an inclusion-exclusion argument that may be viewed as a precursor of the cluster method, applied to the special case of increasing patterns.

Equivalently, we have the following exponential generating function:

$$\sum_{n=0}^{\infty} a_r(n)\frac{x^n}{n!} = \frac{1}{1-x+\frac{x^r}{r!}-\frac{x^{r+1}}{(r+1)!}+\frac{x^{2r}}{(2r)!}-\frac{x^{2r+1}}{(2r+1)!}+\frac{x^{3r}}{(3r)!}-\frac{x^{3r+1}}{(3r+1)!}+\cdots} \quad .$$

While this 'explicit' (exponential) generating function is 'nice', it is more efficient to use the fast recurrence. And indeed, the OEIS has these sequences for $3\le r\le 9$, with many terms. These are (in order): A001212, A117158, A177523, A177533, A177553, A230051, A230231.

2.3.2 Efficient computations of permutations of words with two occurrences of each letter

Let $b_r(n)$ be the number of words with 2 occurrences of each of $1,2,\ldots,n$ avoiding the pattern $1\cdots r$. Quite a few of them are currently (April 17, 2018) in the OEIS, but with relatively few terms

- $b_3(n)$: A177555 (15 terms)
- $b_4(n)$: A177558 (15 terms)
- $b_5(n)$: A177564 (14 terms)
- $b_6(n)$: A177574 (14 terms)
- $b_7(n)$: A177594 (14 terms)

$b_r(n)$ for $r > 7$ are not yet (April 17, 2018) in the OEIS.

We can compute $b_r(n)$ in cubic time as follows. If you plug-in $f_r(2^n)$ into the Fundamental Recurrence, you are forced to consider the more general quantities of the form $f_r(2^\alpha 1^\beta)$. Defining

$$B_r(\alpha, \beta) = f_r(2^\alpha 1^\beta) \quad ,$$

and using symmetry, we get the following recurrence for $B_r(\alpha, \beta)$.

$$B_r(\alpha, \beta) = \alpha B_r(\alpha-1, \beta+1) + \beta B_r(\alpha, \beta-1)$$

$$- \sum_{i_1+i_2=r} \binom{\alpha}{i_1}\binom{\beta}{i_2} B_r(\alpha-i_1, \beta-i_2+i_1) + \sum_{i_1+i_2=r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2} B_r(\alpha-i_1, \beta-i_2+i_1)$$

$$- \sum_{i_1+i_2=2r} \binom{\alpha}{i_1}\binom{\beta}{i_2} B_r(\alpha-i_1, \beta-i_2+i_1) + \sum_{i_1+i_2=2r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2} B_r(\alpha-i_1, \beta-i_2+i_1) - \cdots \quad .$$

In particular $b_r(n) = B_r(n, 0)$. Using this recurrence we (easily!) obtained 80 terms of each of the sequences $b_r(n)$ for $3 \le r \le 9$, and could get many more. See the output file

`http://sites.math.rutgers.edu/~zeilberg/tokhniot/oICPW1.txt` .

2.3.3 Efficient computations of permutations of words with three occurrences of each letter

Let $c_r(n)$ be the number of words with 3 occurrences of each of $1, 2, \dots, n$ avoiding the pattern $1 \cdots r$. Quite a few of them are currently (April 17, 2018) in the OEIS, but with relatively few terms

- $c_3(n)$: A177596 (Only 10 terms)
- $c_4(n)$: A177599 (Only 10 terms)
- $c_5(n)$: A177605 (Only 10 terms)
- $c_6(n)$: A177615 (Only 9 terms)
- $c_7(n)$: A177635 (Only 9 terms)

$c_r(n)$ for $r > 7$ are not yet in the OEIS.

We can compute $c_r(n)$ in quartic time as follows. If you plug-in $f_r(3^n)$ into the Fundamental Recurrence, you are forced to consider the more general quantities of the form $f_r(3^\alpha 2^\beta 1^\gamma)$. Defining

$$C_r(\alpha, \beta, \gamma) = f_r(3^\alpha 2^\beta 1^\gamma) \quad ,$$

and using symmetry, we get the following recurrence for $C_r(\alpha, \beta, \gamma)$.

$$C_r(\alpha, \beta, \gamma) = \alpha C_r(\alpha-1, \beta+1, \gamma) + \beta C_r(\alpha, \beta-1, \gamma+1) + \gamma C_r(\alpha, \beta, \gamma-1)$$

$$- \sum_{i_1+i_2+i_3=r} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3} C_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2)$$

$$+ \sum_{i_1+i_2+i_3=r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3} C_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2)$$

$$- \sum_{i_1+i_2+i_3=2r} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3} C_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2)$$

$$+ \sum_{i_1+i_2+i_3=2r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3} C_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2) - \cdots$$

In particular, $c_r(n) = C_r(n,0,0)$. Using this recurrence we (easily!) obtained 40 terms of each of the sequences $c_r(n)$ for $3 \leq r \leq 9$, and could get many more. See the output file

`http://sites.math.rutgers.edu/~zeilberg/tokhniot/oICPW1.txt` .

2.3.4 Efficient computations of permutations of words with four occurrences of each letter

Let $d_r(n)$ be the number of words with 4 occurrences of each of $1, 2, \ldots, n$ avoiding the pattern $1 \cdots r$. Quite a few of them are currently (April 17, 2018) in the OEIS, but with relatively few terms.

- $d_3(n)$: A177637 (8 terms)
- $d_4(n)$: A177640 (8 terms)
- $d_5(n)$: A177646 (8 terms)
- $d_6(n)$: A177656 (8 terms)
- $d_7(n)$: A177676 (8 terms)

$d_r(n)$ for $r > 7$ are not yet in the OEIS.

We can compute $d_r(n)$ in quintic time as follows. If you plug-in $f_r(4^n)$ into the Fundamental Recurrence, you are forced to consider the more general quantities of the form $f_r(4^\alpha 3^\beta 2^\gamma 1^\delta)$. Defining

$$D_r(\alpha,\beta,\gamma,\delta) = f_r(4^\alpha 3^\beta 2^\gamma 1^\delta) \quad ,$$

and using symmetry, we get the following recurrence for $D_r(\alpha,\beta,\gamma,\delta)$.

$$D_r(\alpha,\beta,\gamma,\delta) = \alpha D_r(\alpha-1,\beta+1,\gamma,\delta)+\beta D_r(\alpha,\beta-1,\gamma+1,\delta)+\gamma D_r(\alpha,\beta,\gamma-1,\delta+1)+\delta D_r(\alpha,\beta,\gamma,\delta-1)$$

$$- \sum_{i_1+i_2+i_3+i_4=r} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3}\binom{\delta}{i_4} D_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2,\delta-i_4+i_3)$$

$$+ \sum_{i_1+i_2+i_3+i_4=r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3}\binom{\delta}{i_4} D_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2,\delta-i_4+i_3)$$

$$- \sum_{i_1+i_2+i_3+i_4=2r} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3}\binom{\delta}{i_4} D_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2,\delta-i_4+i_3)$$

$$+\sum_{i_1+i_2+i_3+i_4=2r+1} \binom{\alpha}{i_1}\binom{\beta}{i_2}\binom{\gamma}{i_3}\binom{\delta}{i_4} D_r(\alpha-i_1,\beta-i_2+i_1,\gamma-i_3+i_2,\delta-i_4+i_3) - \cdots$$

In particular $d_r(n) = D_r(n,0,0,0)$. Using this recurrence we (easily!) obtained 20 terms of each of the sequences $c_d(n)$ for $3 \le r \le 9$, and could get many more. See the output file `http://sites.math.rutgers.edu/~zeilberg/tokhniot/oICPW1.txt` .

Comment: Ira Gessel kindly informed us that an alternative approach to extracting coefficients from the generating function in Theorem 1 is to use the elegant method described in section 3 of his paper on symmetric functions and P-recursiveness [7].

2.4 Keeping track of the number of occurrences

Above we showed how to enumerate words *avoiding* the consecutive pattern $1\cdots r$, in other words, the number of words, with a specified number of each letters, with **zero** such patterns. With a little more effort we can answer the more general question about the number of such words with exactly j consecutive patterns $1\cdots r$ for *any* j, not just $j=0$. Let $W(\mathbf{m}) = W(m_1,\dots,m_n)$ be the set of words in the alphabet $1,\dots,n$ with m_1 1's, ... , m_n n's (note that the number of elements of $W(\mathbf{m})$ is $(m_1+\cdots+m_n)!/(m_1!\cdots m_n!)$).

We are interested in the polynomials in t

$$g_r(\mathbf{m};t) = \sum_{w\in W(\mathbf{m})} t^{\alpha(w)} \quad ,$$

where $\alpha(w)$ is the number of occurrences of the consecutive pattern $1\cdots r$ in the word w. (For example $\alpha(831456178) = 3$ if $r=3$. Note that $\alpha(w)=0$ if w avoids the pattern.)

[Also note that $g_r(\mathbf{m};0) = f_r(\mathbf{m})$ and $g_r(\mathbf{m};1) = (m_1+\cdots+m_n)!/(m_1!\cdots m_n!)$.]

Using GJpats.txt we were able to conjecture the following theorem, whose proof will be presented later.

We first need to define certain families of polynomial sequences.

Definition 2. For any integer $k \ge 1$ and $r \ge 2$, $P_k^{(r)}(t)$ is defined as follows.

If $k<r$, then it is 0. If $k=r$ then it is $t-1$, and if $k>r$ then

$$P_k^{(r)}(t) \;=\; (t-1)\sum_{i=1}^{r-1} P_{k-i}^{(r)}(t) \quad .$$

Theorem 3. *For $k\ge 1, r\ge 2$, the generating function of $g_r(\mathbf{m};t)$, let us call it $G_r(x_1,\dots,x_n;t)$, is*

$$G_r(x_1,\dots,x_n;t) = \frac{1}{1-e_1-\sum_{k=r}^{n} P_k^{(r)}(t)e_k} \quad .$$

This implies the

Fundamental Recurrence for g_r: Let $g_r(\mathbf{m};t)$ be the weight-enumerator of words in the alphabet $\{1,\ldots,n\}$ with m_1 1's, m_2 2's, $\ldots m_n$ n's (where we abbreviate $\mathbf{m} = (m_1,\ldots,m_n)$), using the weight "$t$ raised to the power of the number of occurrences of the consecutive pattern $1\cdots r$".

Also, let V_k be the set of $0-1$ vectors of length n with k ones. Then we have

$$g_r(\mathbf{m}) = \sum_{\mathbf{v}\in V_1} g_r(\mathbf{m}-\mathbf{v}) \, + \sum_{k=r}^{n} \sum_{\mathbf{v}\in V_k} P_k^{(r)}(t)\, g_r(\mathbf{m}-\mathbf{v}) \quad .$$

Analogously to the avoidance case we can get efficient recurrences for the permutations, and words in $1^s \cdots n^s$, for each $s > 1$. For each s it is still polynomial time, but things are slower because of the variable t. This is implemented in the Maple package `ICPWt.txt` .

3 Proofs

3.1 Proof of Theorem 1

We will use the general set-up of the Goulden-Jackson cluster method as described in Noonan and Zeilberger's paper [11], but will be able to make things simpler by taking advantage of the specific structure of our forbidden patterns, which are the increasing patterns $1\cdots r$. That will enable us to use an elegant combinatorial argument, without solving a system of linear equations.

First let us quickly review some basic definitions. (We will not go into the details of the cluster method but readers who wish to see an excellent and concise summary of the cluster method are welcome to refer to the first section of Wen's paper [14].) A **marked word** is a word with some of its factors (consecutive subwords) marked. We are assuming that all the marks are in the set of bad words B. For example (13212; [1,3]) is a marked word with 132 marked, with [1,3] denoting the location of the mark. A **cluster** is a marked word where the adjacent marks overlap with each other and all the letters in the underlying word belong to at least one mark of the cluster. For example (145632; [1,3],[2,4],[4,6]) is a cluster whereas (145632; [1,3],[4,6]) is not. We let the weight of a marked word $w = w_1 w_2 \cdots w_k$ be $weight(w) := (-1)^{|S|} \cdot \prod_{i=1}^{k} x[w_i]$ where S is the set of marks in w. For example, the weight of the cluster (135632; [1,3],[2,4],[4,6]) is $(-1)^3 x_1 x_2 x_3^2 x_5 x_6$.

Let M be the set of all marked words in the alphabet $\{1,\ldots,n\}$. Recall from [11] that $weight(M) = 1 + weight(M)\cdot(x_1 + x_2 + \cdots + x_n) + weight(M)\cdot weight(C)$ where C is the set of all possible clusters. This implies that the multivariate generating function for words avoiding the consecutive pattern $1\cdots r$ (i.e., our target generating function) is equal to $weight(M) = \frac{1}{1-e_1-weight(C)}$. So we only need to figure out $weight(C)$. However, to use the classical Goulden-Jackson cluster method, we would have to solve a system of $\binom{n}{r}$ (the number of bad words) equations (recall that we write C as a summation of $C[v]$'s where v is a word in B, and for each $C[v]$ there is an equation) and no obvious symmetry argument seems to help. So we will use a slick combinatorial approach.

Notice that since the pattern to be avoided is $12 \cdots r$, the clusters can only be of the form

$$(a_1 \cdots a_j; [1, r], \ldots)$$

where

$$1 \leq a_1 < a_2 < \cdots < a_j \leq n \quad .$$

Therefore $weight(C)$ is a summations of multivariate monomials on $x_1, x_2, \ldots, x_n$ where the exponent of each variable x_i is zero or one.

Any fixed monomial in $weight(C)$ can come from many different clusters. The number of clusters it comes from and the coefficient of the monomial are uniquely determined by the number of variables in the monomial. For example, for $r = 3$, the monomial $x_1x_3x_5x_6x_7$ can come from the cluster $(13567; [1, 3], [2, 4], [3, 5])$ or $(13567; [1, 3], [3, 5])$. The first cluster contributes weight $(-1)^3x_1x_3x_5x_6x_7$ whereas the second cluster contributes weight $(-1)^2x_1x_3x_5x_6x_7$. So when summing up, they cancel each other out and there is no monomial $x_1x_3x_5x_6x_7$ in $weight(C)$. So is the case with any other monomial of degree 5. Therefore, let us focus on the monomial $x_1x_2x_3 \cdots x_k$ and figure out its coefficient.

Definition 4. Let $\text{coeff}(x_1x_2 \cdots x_k)$ $(k \geq 1)$ be the coefficient of $x_1x_2 \cdots x_k$ in $weight(C)$.

It is clear that for $k < r$, $\text{coeff}(x_1x_2x_3 \cdots x_k) = 0$, because $12 \cdots k$ cannot be a cluster (it does not have enough letters to be marked). And when $k = r$, we have $\text{coeff}(x_1x_2 \cdots x_k) = -1$, since there can be only one mark. So let us move on to the case when $k > r$. We have the following Lemma.

Lemma 5. *For $k > r$, $coeff(x_1x_2 \cdots x_k) = -\ coeff(x_2x_3 \cdots x_k) -\ coeff(x_3x_4 \cdots x_k) - \cdots - coeff(x_rx_{r+1} \cdots x_k)$. (Equivalently, $coeff(x_1x_2 \cdots x_k) = -\ coeff(x_1x_2 \cdots x_{k-1}) -\ coeff(x_1x_2 \cdots x_{k-2}) - \cdots -\ coeff(x_1x_2 \cdots x_{k-r+1})$.)*

This is because there are $(r - 1)$ ways in which the left-most marked word can "interface" with the one to its immediate right. For example, if the clusters are of the form $(1 \cdots k; [1, r], [3, r + 2], \ldots)$ (that is, the second mark starts at 3), then the contribution will be $(-1)\cdot$ $\text{coeff}(x_3x_4 \cdots x_k)$. This is simply because of the bijection between the set of clusters in the form of $(1 \cdots k; [1, r], [3, r + 2], \ldots)$ with set of the clusters in the form $(3 \cdots k; [3, r + 2], \ldots)$. By peeling off the first mark $[1, r]$, we just lose a factor of (-1) in the coefficient of our monomial.

Similarly, if the clusters are of the form $(1 \cdots k; [1, r], [u, u + r - 1], \ldots)$ $(1 < u \leq r)$, then the contribution from this case will be $(-1)\cdot$ $\text{coeff}(x_ux_{u+1} \cdots x_k)$. Note that if $k < 2r-1$, there cannot be as many as $(r - 1)$ cases. However, in this case, we can make the convention that there are $(r - 1)$ places for the second mark because for $k < r$ the coefficient of $x_1x_2x_3 \cdots x_k$ is 0. So the above formula still holds. For example, for the clusters associated with the word 123456, and $r = 4$, the first mark has to be 1234, the second mark can only be 2345 or 3456. But, according to the natural convention, the second mark can also start with 4 and be 456, and so, $\text{coeff}(x_1x_2x_3x_4x_5x_6) = -\text{coeff}(x_2x_3x_4x_5x_6) - \text{coeff}(x_3x_4x_5x_6) - \text{coeff}(x_4x_5x_6) = -\text{coeff}(x_2x_3x_4x_5x_6) - \text{coeff}(x_3x_4x_5x_6)$.

So we have: $\mathrm{coeff}(x_1x_2\cdots x_r) = -1$; $\mathrm{coeff}(x_1x_2\cdots x_{r+1}) = (-1)\cdot(-1) = 1$; $\mathrm{coeff}(x_1x_2\cdots x_{r+2}) = -\mathrm{coeff}(x_2x_3\cdots x_{r+2}) - \mathrm{coeff}(x_3x_4\cdots x_{r+2}) = -\mathrm{coeff}(x_1x_2\cdots x_{r+1}) - \mathrm{coeff}(x_1x_2\cdots x_r) = 0$. Continuing this process, it is easy to see that $x_1x_2\cdots x_{mr}$ ($m \geq 1$) has coefficient -1 (so is any other monomial of degree mr) and $x_1x_2\cdots x_{mr+1}$ has coefficient 1 (so is any other monomial of degree $mr+1$). The monomials with other number of variables all have coefficient 0. From this argument and summing over all clusters, we conclude $weight(C) = -e_r + e_{r+1} - e_{2r} + e_{2r+1} + \cdots$ and therefore $weight(M) = \frac{1}{1-e_1+e_r-e_{r+1}+e_{2r}-e_{2r+1}+\cdots}$.

3.2 Proof of Theorem 3

This proof can be directly generalized from the proof of Theorem 1 based on the 't-generalization' described in Noonan and Zeilberger's paper [11]. Again, let the set of marked words on $\{1, 2, \ldots, n\}$ be M. However, this time we let the weight of a marked word w of length k be $weight(w) := (t-1)^{|S|} \cdot \prod_{i=1}^{k} x[w_i]$ where S is the set of marks in w. We still have $weight(M) = 1+weight(M)\cdot(x_1+x_2+\cdots+x_n)+weight(M)\cdot weight(C)$ and $G_r(x_1, \ldots, x_n; t)$ is equal to $weight(M)$, which is $\frac{1}{1-e_1-weight(C)}$.

The procedure to calculate $weight(C)$ directly follows from the proof of Theorem 1. We simply replace (-1) with $(t-1)$ in various places, because the only difference is that now we assign a different weight to a marked word. For example, we have $\mathrm{coeff}(x_1x_2\cdots x_r) = t-1$; $\mathrm{coeff}(x_1x_2\cdots x_{r+1}) = (t-1)(t-1) = (t-1)^2$; $\mathrm{coeff}(x_1x_2\cdots x_{r+2}) = (t-1)(\mathrm{coeff}(x_2x_3\cdots x_{r+2}) + \mathrm{coeff}(x_3x_4\cdots x_{r+2})) = (t-1)((t-1) + (t-1)^2)$. Again it is clear that for $k < r$, $\mathrm{coeff}(x_1x_2x_3\cdots x_k) = 0$ and when $k = r$, $\mathrm{coeff}(x_1x_2\cdots x_k) = t-1$. For the case when $k > r$, we generalize Lemma 5 to the following:

Lemma 6. *For $k > r$, coeff$(x_1x_2\cdots x_k) = (t-1)$ (coeff$(x_2x_3\cdots x_k)+$ coeff$(x_3x_4\cdots x_k) + \cdots +$coeff$(x_rx_{r+1}\cdots x_k)$). (Equivalently, coeff$(x_1x_2\cdots x_k) = (t-1)$ (coeff$(x_1x_2\cdots x_{k-1})+$ coeff$(x_1x_2\cdots x_{k-2}) + \cdots +$ coeff$(x_1x_2\cdots x_{k-r+1})$).)*

The proof of Lemma 6 directly generalizes from the proof of Lemma 5. Now one mark contributes a factor of $(t-1)$ instead of (-1) to the weight of a marked word. For example, for the clusters associated with the word 123456, and $r = 3$, the first mark has to be 123, the second mark can be 234 or 345. So $\mathrm{coeff}(x_1x_2x_3x_4x_5x_6) = (t-1)(\mathrm{coeff}(x_2x_3x_4x_5x_6)+\mathrm{coeff}(x_3x_4x_5x_6))$. In general, like in the proof of Theorem 1, if we are interested in keeping track of the number of appearances of the consecutive pattern $12\cdots r$, then there are $(r-1)$ scenarios of clusters that can give rise to the monomial $x_1x_2\cdots x_k$, depending on where the second mark is. By peeling off the first mark, now we lose a factor of $(t-1)$ instead of (-1) in the coefficient of our monomial.

As the coefficients of the monomials of the same length are the same, Lemma 6 immediately implies that $weight(C) = \sum_{k=r}^{n} P_k^{(r)}(t)e_k$ where $P_k^{(r)}(t)$ satisfies the recurrence

$$P_k^{(r)}(t) \;=\; (t-1)\sum_{i=1}^{r-1} P_{k-i}^{(r)}(t) \quad .$$

(In fact $P_k^{(r)}(t)$ is just a concise way of writing coeff$(x_1 x_2 \cdots x_k)$, where the consecutive pattern of interest is $12 \cdots r$.) From this Theorem 3 follows directly.

4 Acknowledgment

Many thanks are due to Sergi Elizalde for help with the references and to Yonah Biers-Ariel for suggestions on the format of the paper. Also many thanks to Justin Troyka for pointing out that “our” Theorem 1 appeared in Ira Gessel’s PhD thesis.

References

[1] A. Baxter, B. Nakamura, and D. Zeilberger, Automatic generation of theorems and proofs on enumerating consecutive-Wilf classes, in *Advances in Combinatorics: Waterloo Workshop in Computer Algebra, W80, May 26-29, 2011* [Volume in honor of Herbert S. Wilf], edited by Ilias Kotsireas and Eugene Zima, Springer, 2013, pp. 121–138.

[2] A. Burstein, Enumeration of words with forbidden patterns, Ph.D. thesis, University of Pennsylvania, 1998.

[3] A. Burstein and T. Mansour, Words restricted by patterns with at most 2 distinct letters, *Electron. J. Combin.* **9** (2002), #R3.

[4] F. N. Davis and D. E. Barton, *Combinatorial Chance*, Hafner, New York, 1962.

[5] S. Elizalde and M. Noy, Consecutive patterns in permutations, *Adv. Appl. Math.* **30** (2003), 110–125 .

[6] I. M. Gessel, Generating functions and enumeration of sequences, PhD thesis, Massachusetts Institute of Technology, 1977.

[7] I. M. Gessel, Symmetric functions and P-recursiveness, *J. Combin. Theory Ser. A* **53** (1990), 257–285.

[8] I. Goulden and D. Jackson, An inversion theorem for cluster decomposition of sequences with distinguished subsequences, *J. London Math. Soc.(2)* **20** (1979), 567–576.

[9] A. Mendes and J.B. Remmel, Permutations and words counted by consecutive patterns, *Adv. Appl. Math*, **37** (2006), 443–480.

[10] B. Nakamura, Computational approaches to consecutive pattern avoidance in permutations, *Pure Math. Appl.* (PU.M.A.) **22** (2011), 253–268.

[11] J. Noonan and D. Zeilberger, The Goulden-Jackson cluster method: extensions, applications, and implementations, *J. Difference Eq. Appl.* **5** (1999), 355–377.

[12] L. Pudwell, Enumeration schemes for pattern-avoiding words and permutations, Ph.D. thesis, Rutgers University, May 2008, http://faculty.valpo.edu/lpudwell/papers/pudwell_thesis.pdf

[13] N. J. A. Sloane, *The On-Line Encyclopedia of Integer Sequences*, http://oeis.org.

[14] X. Wen, The symbolic Goulden-Jackson cluster method, *J. Difference Eq. Appl.* **11** (2005), 173–179.

2010 *Mathematics Subject Classification*: Primary 05A15; Secondary 05A05, 05–04.
Keywords: permutation, word, consecutive pattern, generating function, efficient computation, Goulden-Jackson cluster method.

(Concerned with sequences:
A049774, A117158, A177523, A177533, A177553, A177555, A177558, A177564, A177574, A177594, A177596, A177599, A177605, A177615, A177635, A177637, A177640, A177646, A177656, A177676, A230051, and A230231.)