An Introduction to Enumeration Schemes

Lara Pudwell
Valparaiso University
Lara.Pudwell@valpo.edu

AWM Workshop
Washington, DC
January 8, 2009

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is `1`●●●●.

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is `1••2•`.

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is `1●●23`.

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is `14•23`.

Reduction

- Given a string of numbers $p = p_1 \cdots p_n$, the reduction of p is the string obtained by replacing the i^{th} smallest number of p with i.
- For example, the reduction of `26745` is `14523`.

Pattern Avoidance in Permutations

- Given $p \in S_n$ and $q \in S_m$, we say p contains q if there are $1 \leq i_1 < \cdots < i_m \leq n$ such that $p_{i_1} \cdots p_{i_m}$ reduces to q. Otherwise, p avoids q.
- $p = 21354$ *contains* 132.
(since 21354 reduces to 132.)
- $p = 21354$ *avoids* 321.
(since p has no decreasing subsequence of length 3.)

Two Questions

Easy: Given $p \in S_n$, what patterns does p contain?

Hard: Given $q \in S_m$,

- Let $S_n(q) = \{p \in S_n \mid p \text{ avoids } q\}$.
- Find an expression for $|S_n(q)|$.

Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

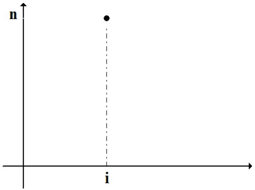
$$|S_n(12)| = |S_n(21)| = 1, n \geq 0$$

Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

$$|S_n(12)| = |S_n(21)| = 1, n \geq 0$$

Graph of $p \in S_n(132)$

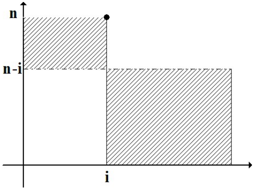


Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

$$|S_n(12)| = |S_n(21)| = 1, n \geq 0$$

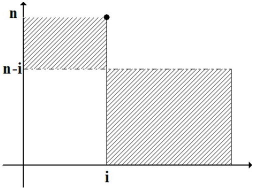
Graph of $p \in S_n(132)$



Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

$$|S_n(12)| = |S_n(21)| = 1, n \geq 0$$



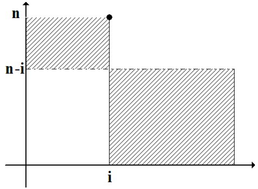
Graph of $p \in S_n(132)$

So, $|S_n(132)| = \sum_{i=1}^{n} |S_{i-1}(132)| \cdot |S_{n-i}(132)|$

Patterns of length 1, 2, and 3

$$|S_n(1)| = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}$$

$$|S_n(12)| = |S_n(21)| = 1, n \geq 0$$



Graph of $p \in S_n(132)$

So, $|S_n(132)| = \sum_{i=1}^{n} |S_{i-1}(132)| \cdot |S_{n-i}(132)| = \dfrac{\binom{2n}{n}}{n+1} = C_n$

In fact, $|S_n(q)| = C_n$ if q is any permutation of length 3.

Patterns of Length 4

There are 24 patterns of length 4.

Using several clever bijections, we can narrow our work to 3 cases:
$S_n(1342), S_n(1234),$ and $S_n(1324)$.

Patterns of Length 4

There are 24 patterns of length 4.

Using several clever bijections, we can narrow our work to 3 cases:
$S_n(1342), S_n(1234),$ and $S_n(1324)$.

	1	2	3	4	5	6	7	8	
$\lvert S_n(1342)\rvert$	1	2	6	23	103	512	2740	15485	$\sim 8^n$
$\lvert S_n(1234)\rvert$	1	2	6	23	103	513	2761	15767	$\sim 9^n$
$\lvert S_n(1324)\rvert$	1	2	6	23	103	513	2762	15793	$\sim 9.3^n$

For Permutations

- Most techniques studying $|S_n(q)|$ finds formulas for a *specific* q.
- 1998: Zeilberger's *prefix enumeration schemes*, i.e. a system of recurrences to count $|S_n(q)|$.
- 2005: Vatter's modified schemes automate the enumeration of $|S_n(q)|$ for even more patterns q.

Refinement Notation

Goal: Divide $S_n(q)$ into subsets.

$$S_n(q; p_1 \cdots p_l) := \left\{ \pi \in S_n \;\middle|\; \begin{array}{l} \pi \text{ avoids } q \\ \pi \text{ has prefix } p_1 \cdots p_l \end{array} \right\}$$

and

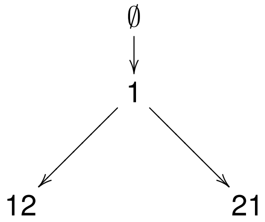
$$S_n\left(q; \begin{array}{c} p_1 \cdots p_l \\ i_1 \cdots i_l \end{array}\right) := \left\{ \pi \in S_n \;\middle|\; \begin{array}{l} \pi \text{ avoids } q \\ \pi \text{ has prefix } p_1 \cdots p_l \\ \pi = i_1 \cdots i_l \pi_{l+1} \cdots \pi_n \end{array} \right\}$$

If $p_1 \cdots p_{l-1}$ reduces to $p_1^* \cdots p_{l-1}^*$, then $p = p_1 \cdots p_l$ is called a refinement if p^*.
e.g $p = 2413$ is a refinement of $p^* = 231$.

Refinement Example

For any pattern q, $S_n(q) = S_n(q;1) = S_n(q;12) \cup S_n(q;21)$.

or graphically:



Reversibly Deletable Positions

Reversibly Deletable Positions

Given a pattern q and a prefix p, position r is reversibly deletable if

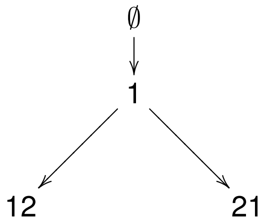
- Deleting p_r from $\pi \in S_n(q; p_1 \cdots p_l)$ produces a q-avoiding permutation of length $n-1$, and
- Inserting p_r into $\pi \in S_{n-1}(q; p_1 \cdots p_{r-1} p_{r+1} \cdots p_l)$ produces a q avoiding permutation of length n.

In other words, deleting and re-inserting p_r gives a bijection between $S_n(q; p_1 \cdots p_l)$ and $S_{n-1}(q; p_1 \cdots p_{r-1} p_{r+1} \cdots p_l)$, and

$$|S_n(q; p_1 \cdots p_l)| = |S_{n-1}(q; p_1 \cdots p_{r-1} p_{r+1} \cdots p_l)| .$$

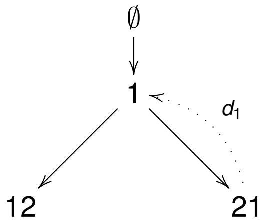
Reversibly Deletable Example

Graphically:



Reversibly Deletable Example

Graphically:

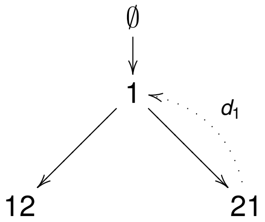


Gap Vectors

Gap vectors give a condition for which choices of $i_1, \ldots, i_l$ yield

$$\left| S_n \left(q; \begin{matrix} p_1 \cdots p_r \cdots p_l \\ i_1 \cdots i_r \cdots i_l \end{matrix} \right) \right| = 0.$$

Graphically:

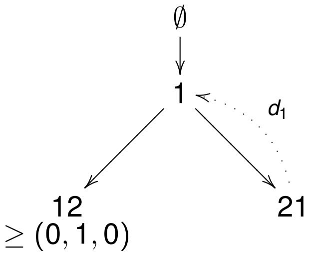


Gap Vectors

Gap vectors give a condition for which choices of $i_1, \ldots, i_l$ yield

$$\left| S_n \left(q; \begin{matrix} p_1 \cdots p_r \cdots p_l \\ i_1 \cdots i_r \cdots i_l \end{matrix} \right) \right| = 0.$$

Graphically:

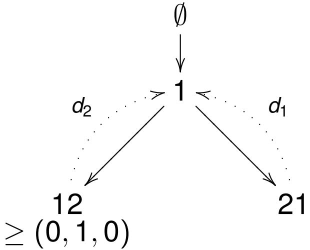


Gap Vectors

Gap vectors give a condition for which choices of $i_1, \ldots, i_l$ yield

$$\left| S_n \left(q; \begin{matrix} p_1 \cdots p_r \cdots p_l \\ i_1 \cdots i_r \cdots i_l \end{matrix} \right) \right| = 0.$$

Graphically:



Enumeration Scheme Definition

An enumeration scheme is a set of triples $[p_i, R_i, G_i]$ such that for each triple

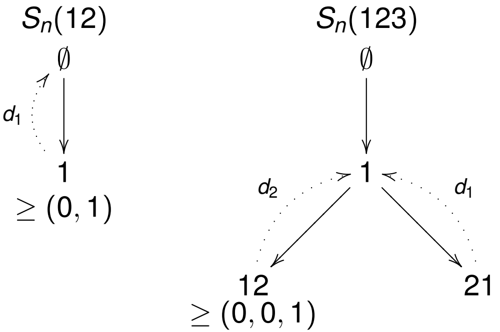
- p_i is a reduced prefix of length n
- R_i a subset of $\{1, \ldots, n\}$
- G_i is a set of vectors of length $n+1$
  and
- either R_i is non-empty or all refinements of p_i are also in the scheme.

Enumeration Scheme Definition

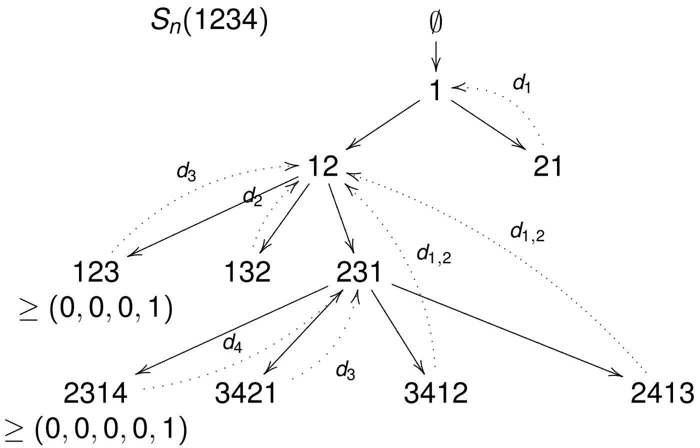
An enumeration scheme is a set of triples $[p_i, R_i, G_i]$ such that for each triple

- p_i is a reduced prefix of length n (prefix)
- R_i a subset of $\{1, \ldots, n\}$ (reversibly deletable positions)
- G_i is a set of vectors of length $n + 1$ (gap vectors) and
- either R_i is non-empty or all refinements of p_i are also in the scheme.

Avoid(12) and Avoid(123)



Avoid(1234)


$S_n(1234)$
∅
1
d_1
12
21
d_3
d_2
$d_{1,2}$
$d_{1,2}$
123
132
231
$\geq (0,0,0,1)$
d_4
d_3
2314
3421
3412
2413
$\geq (0,0,0,0,1)$

Summary

- There are few techniques to count large classes of pattern-avoiding permutations.
- Extending Zeilberger's and Vatter's schemes gives a good success rate for counting the elements of $S_n(q)$.

- Enumeration schemes have also been successfully used to count:
  - pattern-avoiding words (strings with repeated letters)
  - permutations avoiding barred patterns (permutations that avoid a particular pattern unless that pattern is part of an even larger specified pattern)

Contact Info

Lara Pudwell
Lara.Pudwell@valpo.edu
http://faculty.valpo.edu/lpudwell