Valparaiso University

Automated Enumeration of Pattern Avoiding Permutations

Lara Pudwell
Valparaiso University

Pacific Workshop on Pattern Avoiding Permutations
March 31, 2011

Goals

Goal

Find algorithmic ways to study $S_n(Q)$.

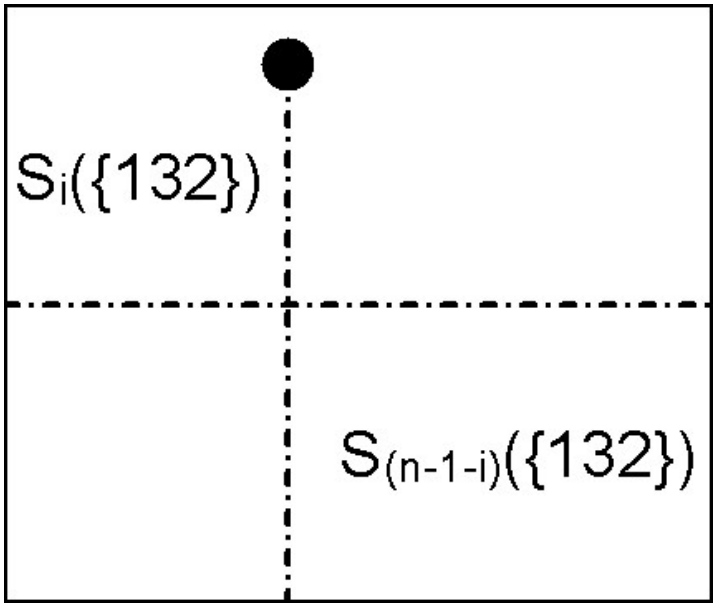
Subgoal

Find enumeration techniques that apply to a wide variety of pattern sets.

Why?

- Produce exact enumeration results.
- Generate data to form further conjectures.

Substitution Decomposition



Theorem (Albert and Atkinson 2005)

If $S_n(Q)$ contains finitely many simple permutations, then has an algebraic generating function.

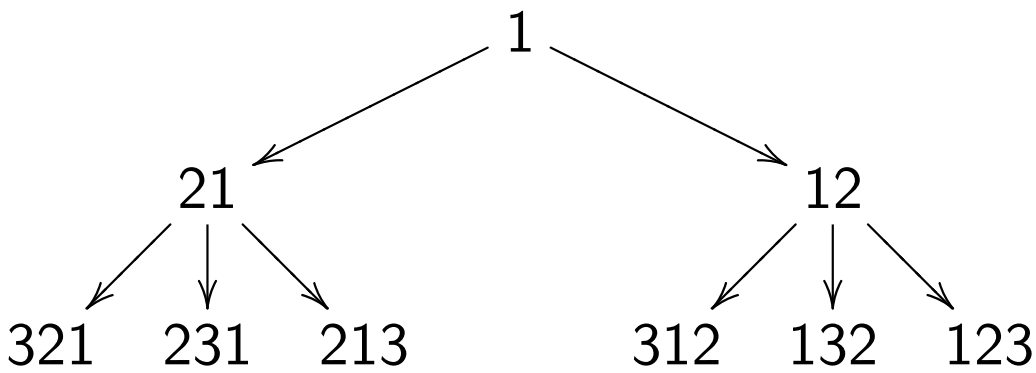
Insertion Encoding

- Associates to each permutation a word describing how that permutation evolved.
- If the insertion encoding for $S_n(Q)$ forms a regular language then $S_n(Q)$ has a rational generating function that can be routinely computed.

Theorem (Albert, Linton, and Ruškuc 2005)

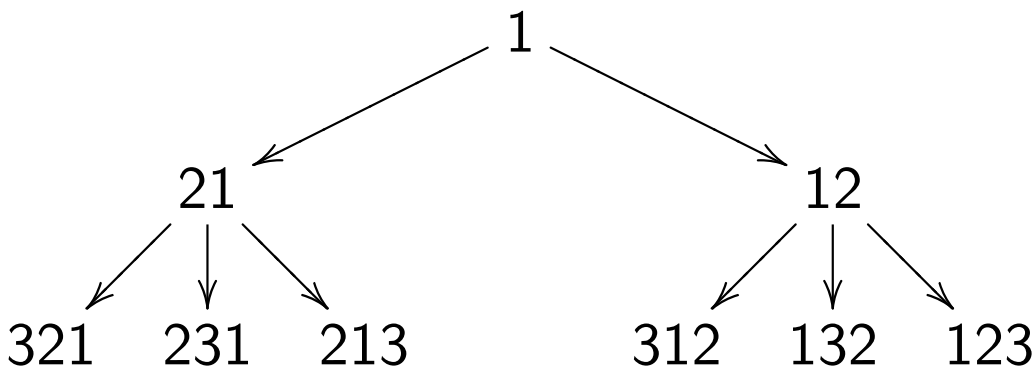
The insertion encoding for $S_n(Q)$ forms a regular language if and only if $S_n(Q)$ contains only finitely many vertical alternations.

Generating Tree Example



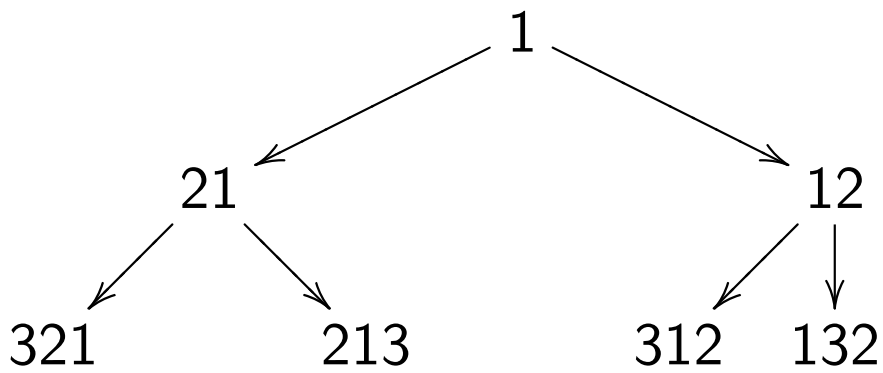
Generating Tree Example

Avoiding $\{123, 231\}$



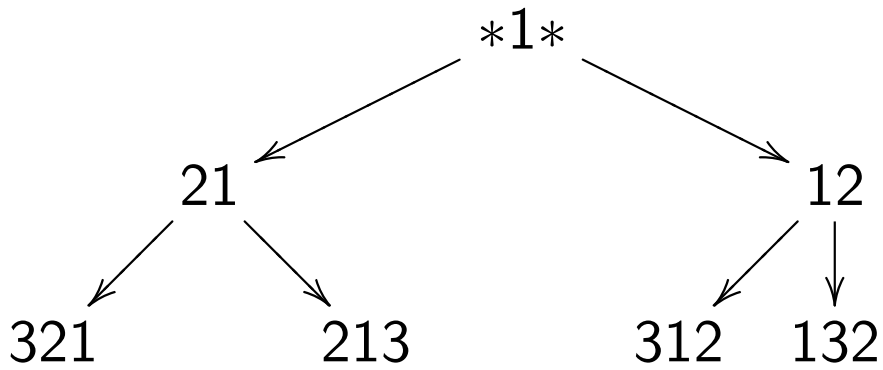
Generating Tree Example

Avoiding $\{123, 231\}$



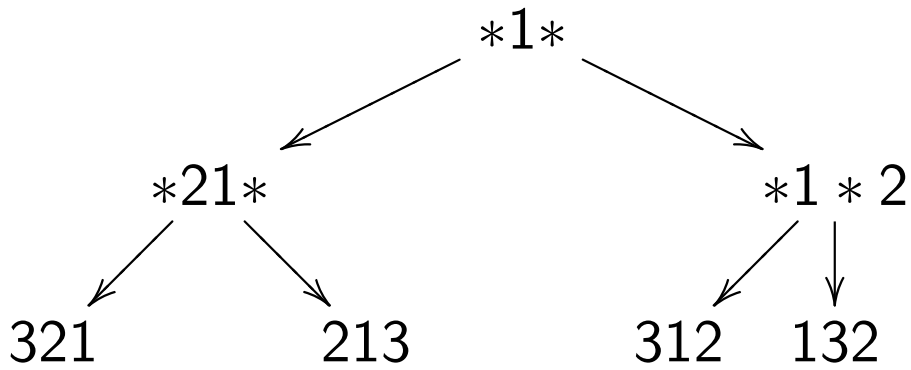
Generating Tree Example

Avoiding $\{123, 231\}$



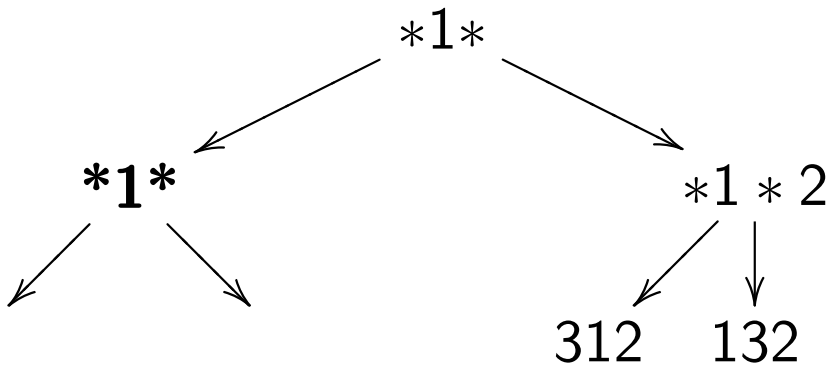
Generating Tree Example

Avoiding $\{123, 231\}$



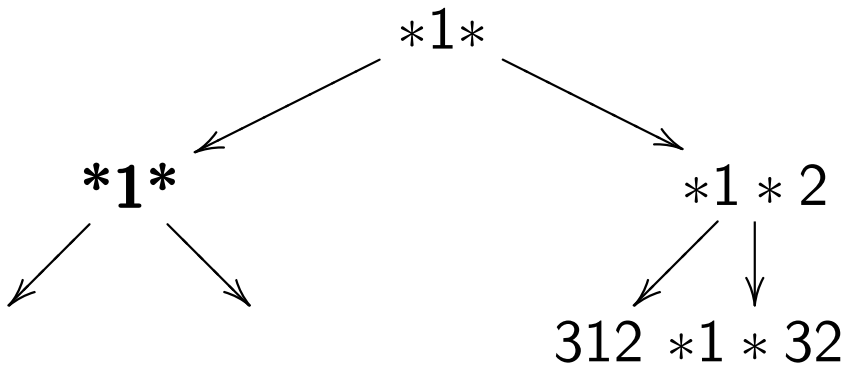
Generating Tree Example

Avoiding $\{123, 231\}$



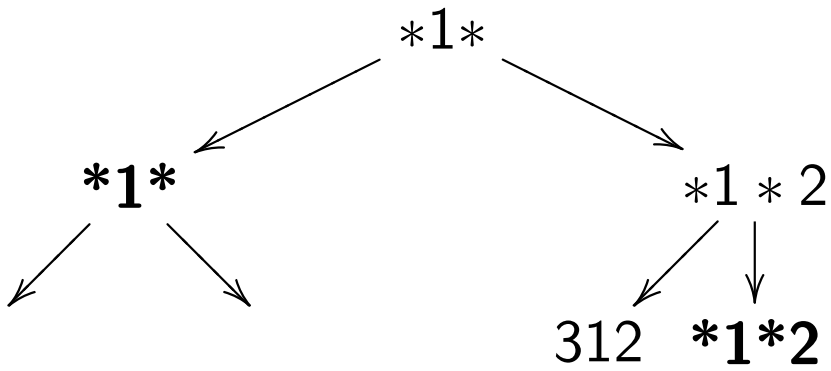
Generating Tree Example

Avoiding $\{123, 231\}$



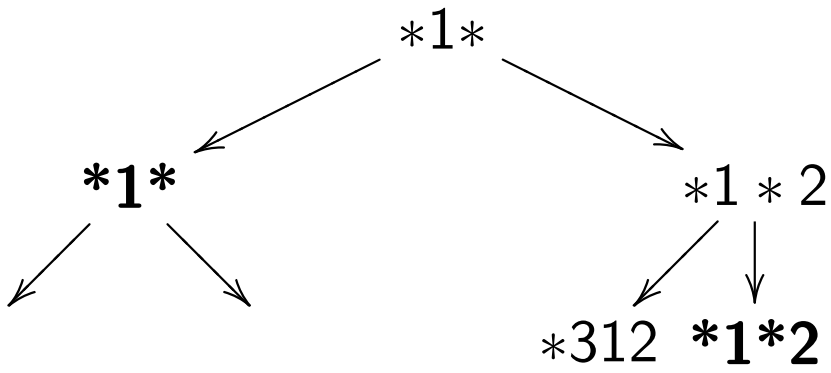
Generating Tree Example

Avoiding $\{123, 231\}$



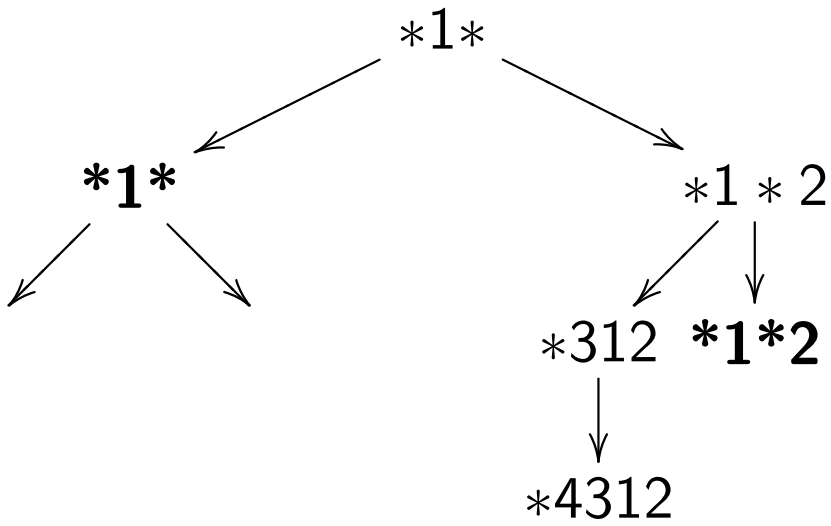
Generating Tree Example

Avoiding $\{123, 231\}$



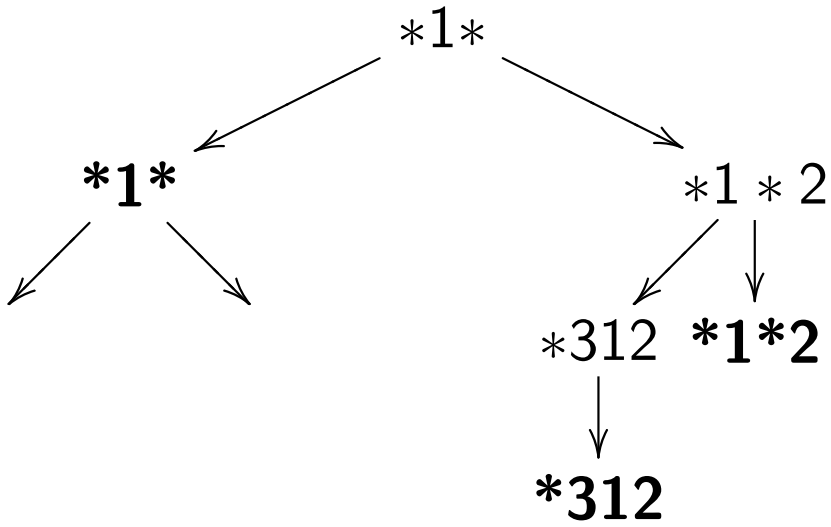
Generating Tree Example

Avoiding $\{123, 231\}$



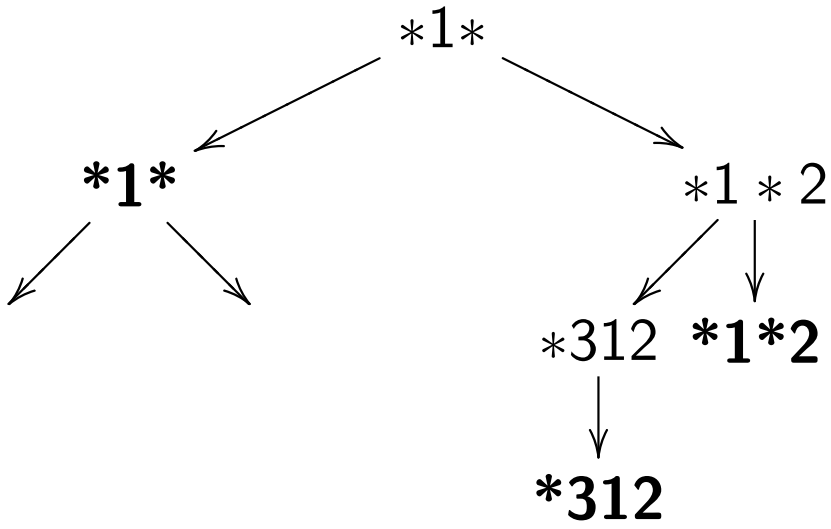
Generating Tree Example

Avoiding $\{123, 231\}$



Generating Tree Example

Avoiding $\{123, 231\}$



Rules:

$1 \to 1, 12$

$12 \to 12, 312$

$312 \to 312$

Generating Tree Example

Avoiding $\{123, 231\}$

Rules:

$1 \to 1, 12$

$12 \to 12, 312$

$312 \to 312$

Transfer Matrix...

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Utility of Generating Trees

- Focuses on placement of smallest letters in a permutation.
- Easy to compute $|S_n(Q)|$ from transfer matrix for specific n.
- Easy to compute generating function from transfer matrix.
- Can be completely automated by computer.

Theorem (Vatter 2006)

$S_n(Q)$ corresponds to a finitely labeled generating tree if and only if Q contains both a child of an increasing permutation and a child of a decreasing permutation.

Enumeration Scheme Definition/Notation

Definition (informal)

An *enumeration scheme* is an encoding for a family of recurrence relations enumerating members of a family of sets.

Definition: prefix pattern

$$S_n(Q)[p] = \{\pi \in S_n(Q) | \pi_1 \cdots \pi_{|p|} \sim p\}$$

Note

$$S_n(Q) = S_n(Q)[1] = S_n(Q)[12] \cup S_n(Q)[21] = \cdots$$

Enumeration Scheme Notation

Definition: prefix pattern with specified letters

$$S_n(Q)[p; w] = \{\pi \in S_n(Q)[p] | \pi_1 \cdots \pi_{|p|} = w\}$$

Note

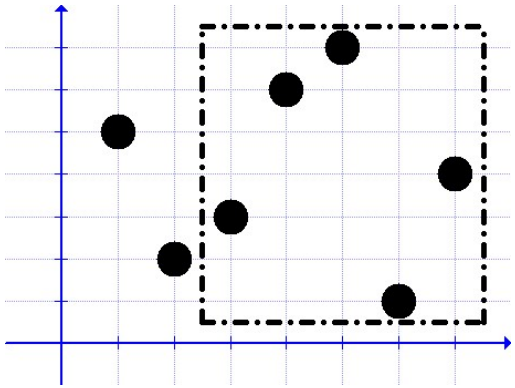
$$\left|S_n(Q)[1]\right| = \sum_{i=1}^{n} \left|S_n(Q)[1; i]\right|$$

Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[21]$...

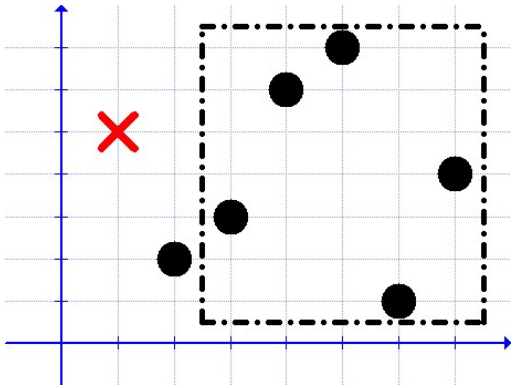


Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[21]$...

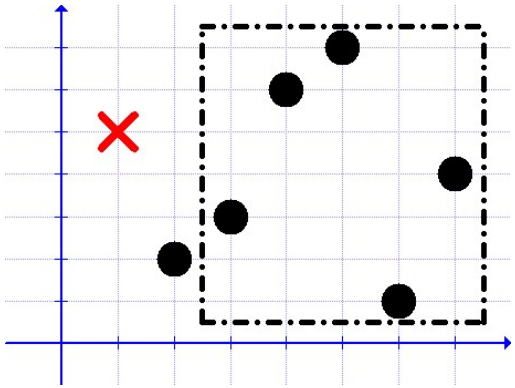


Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[21]$...



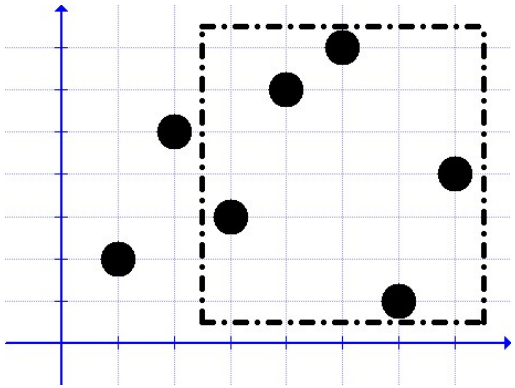
So $\left|S_n(\{123\})[21; ij]\right| = \left|S_{n-1}(\{123\})[1; j]\right|$

Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[12]$...

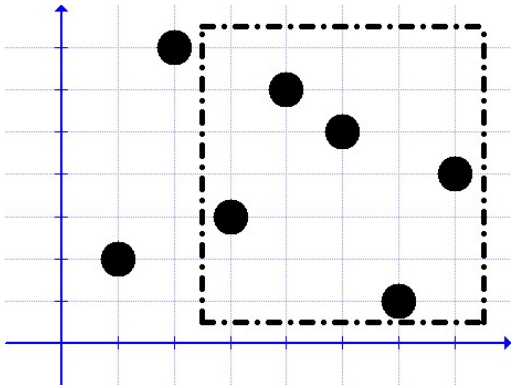


Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[12]$...

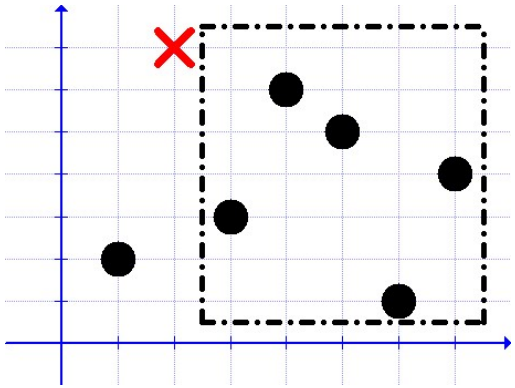


Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[12]$...

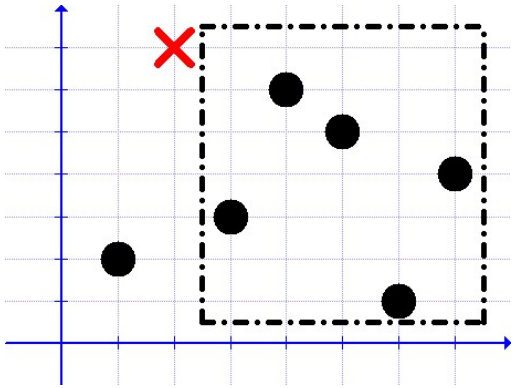


Enumeration Scheme Example

Avoiding $\{123\}$

$$S_n(\{123\}) = S_n(\{123\})[1] = S_n(\{123\})[12] \cup S_n(\{123\})[21]$$

Consider a member of $S_n(\{123\})[12]$...



So $\left|S_n(\{123\})[12; ij]\right| = \begin{cases} 0 & j < n \\ \left|S_{n-1}(\{123\})[1; i]\right| & j = n \end{cases}$

Enumeration Scheme Example

Avoiding $\{123\}$ Summary:

$$\begin{aligned}
\left|S_n(\{123\})\right| &= \left|S_n(\{123\})[1]\right| = \sum_{i=1}^{n} \left|S_n(\{123\})[1;i]\right| \\
\left|S_n(\{123\})[1;i]\right| &= \sum_{j=1}^{i-1} \left|S_n(\{123\})[21;ij]\right| \\
&\quad + \sum_{j=i+1}^{n} \left|S_n(\{123\})[12;ij]\right| \\
\left|S_n(\{123\})[21;ij]\right| &= \left|S_{n-1}(\{123\})[1;j]\right| \\
\left|S_n(\{123\})[12;ij]\right| &= \begin{cases} 0 & j < n \\ \left|S_{n-1}(\{123\})[1;i]\right| & j = n \end{cases}
\end{aligned}$$

Utility of Enumeration Schemes

- Focus on initial few letters in a permutation.
- Work for many more pattern sets (including singletons).
- Can be completely automated by computer.
- Can be transformed into a functional equation satisfied by generating function. (Baxter 2011)
- Can be used to q-count according to inversion number, and according to the number of occurrences of any consecutive pattern(s). (Baxter 2011)
- Extend to many other types of pattern avoidance problems.

Extensions

- Words avoiding permutations (Pudwell 2008)
  - Instead of $S_n(Q)$ consider words with a_1 1's, ..., a_n n's avoiding Q.
- Permutations avoiding barred patterns (Pudwell 2008)
  - e.g. $2\bar{4}153$
- Permutations avoiding dashed patterns (Baxter, Pudwell 2011)
  - e.g. $2-3-14$

Open Questions

1. We have algorithmic techniques to deal with barred patterns and dashed patterns (i.e. lots of data). Can we characterize when they are equivalent and when they are not?
   Examples:
   - $S_n(\{25\bar{3}14\}) = S_n(\{2-41-3\})$
   - There is no dashed pattern equivalent to $2\bar{5}341$.
   - $\left|S_n(\{3\bar{5}241\})\right| = \left|S_n(\{31-4-2\})\right|$ but $S_n(\{3\bar{5}241\}) \neq S_n(\{31-4-2\})$

Open Questions

1. We have algorithmic techniques to deal with barred patterns and dashed patterns (i.e. lots of data). Can we characterize when they are equivalent and when they are not?
   Examples:
   - $S_n(\{25\bar{3}14\}) = S_n(\{2-41-3\})$
   - There is no dashed pattern equivalent to $2\bar{5}341$.
   - $\left|S_n(\{3\bar{5}241\})\right| = \left|S_n(\{31-4-2\})\right|$ but $S_n(\{3\bar{5}241\}) \neq S_n(\{31-4-2\})$
2. Can we find algorithmic techniques to enumerate permutations avoiding bivincular patterns?
3. Can we find algorithmic techniques to study other types of patterns that appear in this workshop?

Thank You!