Pattern Avoidance in Binary Trees

Eric Rowland

Mathematics Department
Tulane University, New Orleans, USA

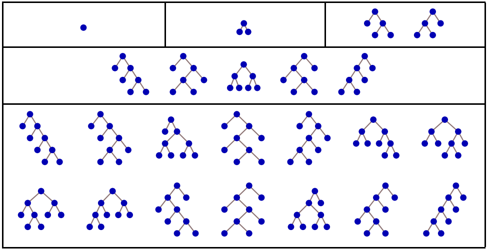
June 22, 2011

Outline

Outline

Binary trees

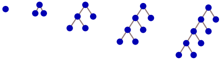
Binary trees with ≤ 5 leaves:

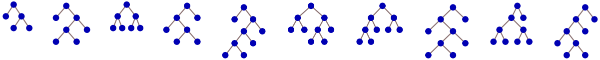


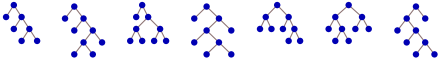
Pattern containment

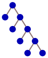
Patterns are contiguous. For example, let $t =$ .

Small binary trees containing...

0 copies of t: 

1 copy of t: 

2 copies of t: 

3 copies of t: 

Outline

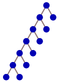
1. Definitions
2. Enumeration
3. Equivalence

Small patterns

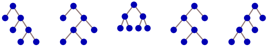
What is the number $a(n)$ of n-leaf binary trees avoiding t?

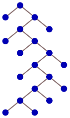
1-leaf tree patterns: $t =$ .
$a(n) = 0$.

2-leaf tree patterns: $t =$ .
$a(1) = 1$; $a(n) = 0$ for $n \geq 2$.

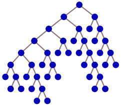
3-leaf tree patterns:  and . “Typical” tree avoiding : .
$a(n) = 1$.

4-leaf tree patterns

4-leaf tree patterns: .

- $t =$ . A “typical” tree avoiding t looks like .

  $a(1) = 1$; $a(n) = 2^{n-2}$ for $n \geq 2$.

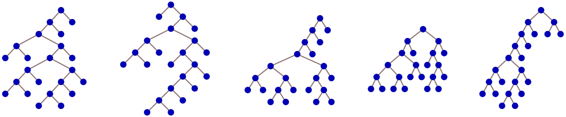
- $t =$ . A “typical” tree avoiding t looks like .

  $a(1) = 1$; $a(n) = 2^{n-2}$ for $n \geq 2$.

These two patterns are equivalent.

The remaining 4-leaf pattern

- $t =$ . Some trees avoiding t:

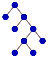


Donaghey and Shapiro showed that $a(n) = M_{n-1}$.

Generating functions algorithm

Let $\mathrm{Av}_t(x) = \displaystyle\sum_{T \text{ avoids } t} x^{\text{number of vertices in } T} = \sum_{n=0}^{\infty} a(n) x^n.$

Consider the tree pattern

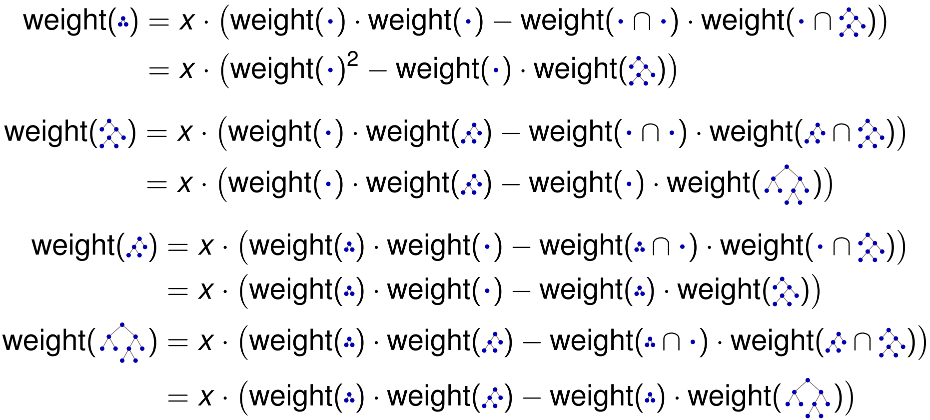
$t =$  $= \overset{\bullet}{t_l\, t_r},$ where $t_l = \bullet$ and $t_r =$ .

For a given tree pattern p, let

$$\mathrm{weight}(p) := \sum_{T \text{ matches } p \text{ and avoids } t} x^{\text{number of vertices in } T}.$$

Begin with $\mathrm{weight}(\cdot) = x + \mathrm{weight}($$)$; rewrite $\mathrm{weight}($$)$ by

$$\mathrm{weight}\big(\overset{\bullet}{p_l\, p_r}\big) = x \cdot \big(\mathrm{weight}(p_l) \cdot \mathrm{weight}(p_r) - \mathrm{weight}(p_l \cap t_l) \cdot \mathrm{weight}(p_r \cap t_r)\big).$$



No new variables. Eliminate the four auxiliary variables to obtain

$$x^3\,\mathrm{Av}_t(x)^2 - (x^2-1)^2\,\mathrm{Av}_t(x) - x\,(x^2-1) = 0.$$

Details

The algorithm terminates:
Since $\mathrm{depth}(p \cap p') \leq \max(\mathrm{depth}(p), \mathrm{depth}(p'))$ and there are only finitely many trees shallower than t, there are finitely many variables.

Final step:
Eliminating all variables except x and $\mathrm{weight}(\cdot) = \mathrm{Av}_t(x)$ entails computing a Gröbner basis for the system, which may take time.

Theorem

$\mathrm{Av}_t(x)$ *is algebraic.*

And the algebraic equation can be constructed mechanically.

Generalizations

The enumerating generating function

$$\mathrm{En}_t(x, y) = \sum_T x^{\text{number of vertices in } T} y^{\text{number of copies of } t \text{ in } T}$$

is algebraic.

Conjecture

If s and t are avoiding-equivalent, then they are also enumerating-equivalent.

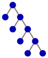
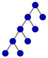
The generating function that enumerates trees with respect to multiple patterns is also algebraic.

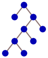
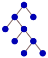
5-leaf equivalence classes

A computer implementation establishes all equivalence classes for binary trees up to 8 leaves.

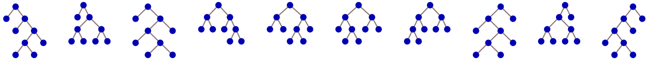
For $m = 1, 2, 3, \ldots$, there are $1, 1, 1, 2, 3, 7, 15, 44, \ldots$ equivalence classes of m-leaf binary trees.

For 5-leaf tree patterns...

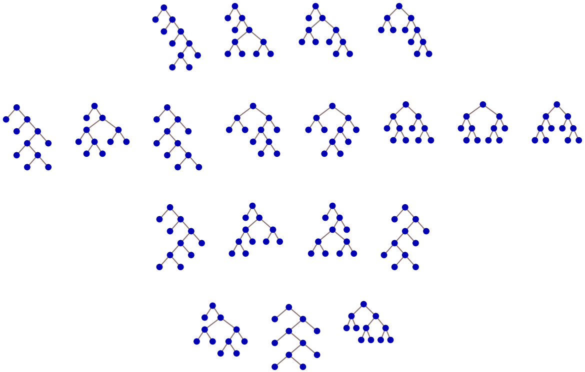
 and  form an equivalence class.

 and  form an equivalence class.

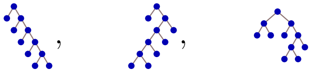
The other 10 tree patterns are equivalent:



6-leaf equivalence classes



Isolated trees:



Outline

Analogy to avoidance in words

Can we easily tell when *s* and *t* are equivalent?

How many length-*n* words on an alphabet avoid a certain (contiguous) subword?

The answer depends only on the self-overlap lengths of the subword.

Equivalence classes of length-4 words on the alphabet $\{0, 1\}$:

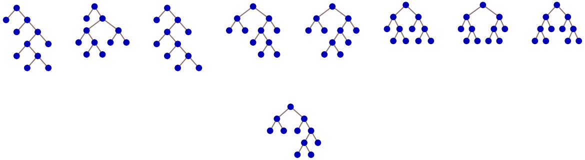
equivalence class	self-overlap lengths
$\{0001, 0011, 0111, 1000, 1100, 1110\}$	$\{4\}$
$\{0010, 0100, 0110, 1001, 1011, 1101\}$	$\{1, 4\}$
$\{0101, 1010\}$	$\{2, 4\}$
$\{0000, 1111\}$	$\{1, 2, 3, 4\}$

Self-overlaps

For trees, one might hope for the same criterion:
"*s* and *t* are equivalent precisely when their self-overlaps coincide."

The self-overlap lengths seem to be preserved under equivalence.

But the statement is not true in general. Counterexample:



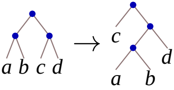
Both classes have overlap lengths $\{1, 1, 2, 2, 5\}$.

Given two equivalent tree patterns *s* and *t*, can we find a bijective proof of the equivalence?

For example,  and  are equivalent.

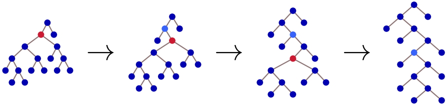
Let *T* avoid . Idea: Replace all instances of  with .

How?



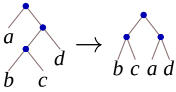
What order? top-down.

For example:

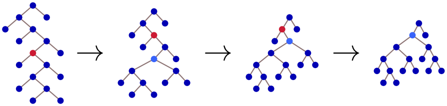


Inverse

The inverse map is a *bottom-up replacement* with the inverse replacement rule:



For example:



Searching for bijections

Generally, we may individually try all $m!$ permutations of leaves.

Permutations whose replacements prove equivalence for pairs of 5-leaf trees:

	t_2	t_3	t_4	t_6	t_7	t_8	t_9	t_{11}	t_{12}	t_{13}
t_2	—	14235		43125						
t_3		—	12534	31245					51234	
t_4		12453	—			41235				
t_6				—	12534	45123				
t_7				12453	—		45123			
t_8				34512		—	31245			
t_9					34512	23145	—			
t_{11}					13452			—	31245	
t_{12}		23451					12453	23145	—	
t_{13}							14532		13425	—

Conjecture

Not every pair of equivalent trees can be shown equivalent by such a bijection. However, the following appears to hold.

Conjecture

Two binary tree patterns s and t are equivalent if and only if there is a sequence of

- *top-down replacements,*
- *bottom-up replacements, and*
- *left–right reflections*

that produces a bijection from binary trees avoiding s to binary trees avoiding t.

The conjecture is true for tree patterns of ≤ 7 leaves.

Open questions

What is the growth rate of the sequence $1, 1, 1, 2, 3, 7, 15, 44, \ldots$ that counts the equivalence classes of m-leaf binary tree patterns for $m = 1, 2, 3, \ldots$?

Is there a simple characterization of the permutations of leaves that establish the equivalence of s and t?

Is there a simple characterization of the binary tree patterns that are equivalent to t?

More recent work

2010 REU students at Valparaiso University:
Nathan Gabriel, Katie Peske, and Sam Tay classified ternary tree patterns with ≤ 9 leaves and introduced a representation of trees as sets of words that provides a new class of bijections.

2011 REU students:
Mike Dairyko, Samantha Tyner, and Casey Wynn are studying avoidance of non-contiguous binary tree patterns.
Work in progress. . .