Non-Consecutive Pattern Avoidance in Binary Trees

Michael Dairyko*
Department of Mathematics
Pomona College
Claremont, CA 91711, USA

Lara Pudwell*
Department of Mathematics
and Computer Science
Valparaiso University
Valparaiso, IN 46383, USA
Lara.Pudwell@valpo.edu

Samantha Tyner*
Department of Mathematics
and Computer Science
Augustana College
Rock Island, IL 61201, USA

Casey Wynn*
Department of Mathematics
and Computer Science
Hendrix College
Conway, AR 72032, USA

July 28, 2011

Abstract

In this paper we consider the enumeration of full binary trees avoiding non-consecutive binary tree patterns. We begin by modifying a known algorithm that counts binary trees avoiding a single consecutive tree pattern. Next, we use our algorithm to prove several theorems about the generating function whose n^{th} coefficient gives the number of n-leaf trees avoiding a pattern. In addition, we investigate and structurally explain the recurrences that arise from these generating functions. Finally, we examine the enumeration of binary trees avoiding multiple tree patterns.

*Partially supported by NSF Grant DMS-0851721

1 Introduction

1.1 Definitions and Notation

In order to discuss the concept of *non-consecutive pattern avoidance in trees*, we must first define a new language.

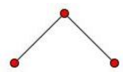
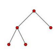
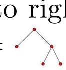


Figure 1: 2-Leaf Tree

The figure above is a *tree* because it is a connected graph with no cycles. It can also be considered a *rooted tree*, or a tree with a special vertex called the root, from which all other vertices branch down. An *ordered tree* is a rooted tree where the left to right order of vertices are important. In particular, for ordered trees  $\neq$ .

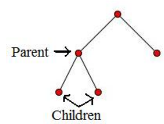
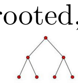
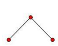
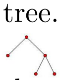
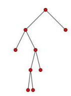
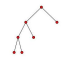


Figure 2: Parent and Children

This figure depicts a *parent* (a vertex that has two edges branching from it) and a *child* (the vertex that is attached to the parent). These definitions lead to a *full binary tree*, or a rooted, ordered tree where every vertex has exactly zero or two children, *e.g.* . Additionally, a *generation* refers to all of the leaves or internal vertices that exist on the same level of a given tree. For example,  has only one generation consisting of 2 leaves, and  has two generations: the first generation consists of the leftmost leaf and the right internal vertex, and the second consists of the two remaining leaves that are the grandchildren of the root. (Note that we have ignored the root in defining the generations of trees, so for completeness, let the root be called the 0^{th} generation.)

As defined by Rowland, *non-consecutive pattern avoidance in trees* is when a binary tree T *avoids* a tree pattern t if there is no instance of t anywhere inside T.

e.g. Let $t_1 =$  and $t_2 =$  .

Notice that t_2 is a 4-leaf left leaning comb, where a *comb* is a tree in which all children lean consecutively to the right or consecutively to the left. Looking at t_1, if the right edge branching from the left child of the root contracts, then the tree becomes a 4-leaf left leaning comb, which contains t_2 consecutively. Therefore we can see that t_1 contains t_2 non-consecutively, as depicted in the following figure,

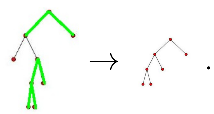


Figure 3: Non-Consecutive Avoidance

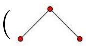
Notation:
Let $Av_n(t)$ be the number of n-leaf trees which avoid the tree pattern t. A generating function is a power series whose coefficients make up a sequence of natural numbers, a_n. We will be particularly interested in the generating function

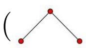
$$gf_t(\cdot) = \sum_{n \geq 0} Av_n(t) \cdot x^n,$$

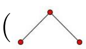
which we use as a tool to better understand $Av_n(t)$ for various trees t.

2 A Generating Function Algorithm

When observing non-consecutive tree patterns, the generating functions are not the same as when avoiding consecutive tree patterns. Therefore, with the help of Rowland's algorithm for consecutive avoidance, we developed a new algorithm that computes generating functions for full binary trees that avoid tree patterns non-consecutively. In order to prove the effectiveness of our algorithm, we first need to introduce new notation. Let $gf_t(p)$ be the generating function for the number of n-leaf full binary trees that avoid the tree pattern t non-consecutively and contain the tree pattern p at their root. Because all binary trees begin with a single vertex, it follows that the generating function for all trees avoiding t is given by $gf_t(\cdot)$. Similarly, let t_ℓ and t_r denote the subtrees descending from the the left and right children of the root of t respectively.

Since we are working with full binary trees, the root has either zero or two children. When there are zero children, we have a 1-leaf tree, which can be represented with the generating function x (*i.e.* $x + 0 \cdot x^2 + 0 \cdot x^3 + \cdots$). When there are two children, we have a tree with two or more leaves which can be denoted by the generating function $gf_t($$)$ which counts the number of trees that avoid t where the root has two children. Thus, our equations are as follows:

$gf_t(\cdot) = x + gf_t($$)$ (1)

$gf_t($$) = gf_{t_\ell}(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf_{t_r}(\cdot) - gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot)$ (2)

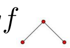
To explain the second equation, we split the counting into two cases:

Case 1: Assume that the subtree extending from the left child of the root avoids t_ℓ. This means there cannot be a copy of t that includes the root or that is to the left of the root so the right subtree need only avoid t. In terms of generating functions, this case can be represented as $gf_{t_\ell}(\cdot) \cdot gf_t(\cdot)$.

Case 2: Similar to Case 1, assume that the subtree extending from the right child of the root avoids t_r. This means there cannot be a copy of t that includes the root or that is to the right of the root so the left subtree need only avoid t. In terms of generating functions, this case can be represented as $gf_t(\cdot) \cdot gf_{t_r}(\cdot)$.

Notice that in the first term of equation (2), $t_r \subset t$ and in the second term $t_\ell \subset t$. Therefore we can see that within the first two terms, we have $2 \cdot gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot)$. In order to resolve this issue of over-counting, we must subtract off $gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot)$.

Our algorithm is as follows,

1. Begin with equation (1) and equation (2)
   using the base cases $gf_{\cdot}(\cdot) = 0$ and gf$(\cdot) = x$
2. Let S be the set of all tree patterns that appear as a subscript on the right hand side of an equation but not the left hand side.
   If $S = \emptyset$ then solve the system of equations for $gf_t(\cdot)$ and end.

3. For each tree pattern $t^* \in S$, compute equations (1) and (2) for t^* and go back to step 2.

Looking at equation (2), notice that any new generating functions that appear on the right hand side are indexed by the subtrees t_ℓ and t_r. By definition, a subtree will contain fewer leaves than the original tree t. There are finitely many trees with fewer leaves than t, thus we will only apply equation (2) finitely many times before the algorithm terminates.

Now we will generalize the equations in our algorithm. Let us combine equations (1) and (2),

$$gf_t(\cdot) = x + gf_{t_\ell}(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf_{t_r}(\cdot) - gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot)$$

Now we can solve for $gf_t(\cdot)$.

$$\begin{aligned} gf_t(\cdot) - gf_{t_\ell}(\cdot) \cdot gf_t(\cdot) - gf_t(\cdot) \cdot gf_{t_r}(\cdot) &= x - gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot) \\ gf_t(\cdot)(1 - gf_{t_\ell}(\cdot) - gf_{t_r}(\cdot)) &= x - gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot) \end{aligned}$$

Thus we see that

$$gf_t(\cdot) = \frac{x - gf_{t_\ell}(\cdot) \cdot gf_{t_r}(\cdot)}{1 - gf_{t_\ell}(\cdot) - gf_{t_r}(\cdot)}. \tag{3}$$

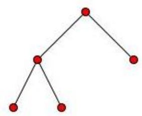
Theorem 1. *For any tree pattern t, $gf_t(\cdot)$ is a rational expression.*

Proof.
Base Case: We know that $gf_t(\cdot)$ is rational for all tree patterns t with $n \leq 2$.
Inductive Step: We assume the theorem is true for all tree patterns with k leaves where $k \leq n$.
Then for any $(n+1)$-leaf tree pattern t, we have a rational combination of rational functions.
$\therefore$ $gf_t(\cdot)$ is rational and by the mathematical principal of induction, the given theorem is true. $\square$

In the next few sections we will use our algorithm to compute generating functions for all 3-,4-, and 5- leaf tree patterns before generalizing our results.

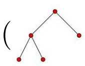
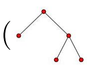
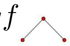
2.1 Avoiding 3-Leaf Trees

Note: The left-right reflections of all n-leaf trees will have the same generating function as the original tree. Therefore to avoid repetition, we will not describe the right leaning reflections.

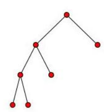
1. Let $t =$ 

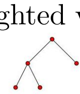
   (a) We see that only one n-leaf tree avoids the tree pattern t, which is the right-leaning n-leaf tree. Thus, $Av_n(t) = 1$ for all $n \leq 1$. Thus,

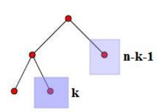
$$gf_t(\cdot) = \frac{x}{1-x}.$$

Note: From this point on we will simplify $Av_n($  $)$ and $Av_n($  $)$ to 1, as well as $gf_{\cdot}(\cdot) = 0$ and gf  $(\cdot) = x$ after the case-by-case breakdown of avoidance.

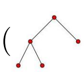
2.2 Avoiding 4-Leaf Trees

1. Let $t =$ 

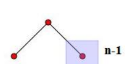
   (a) *Case* 1: Here we assume that the left subtree from the root must have two children, and the leftmost child must be a leaf. Then we know the highlighted vertex in the second generation is a k-leaf tree that must avoid , since it already is connected to the left child of the root, and the right child of the root is a $(n-k-1)$-leaf tree that must simply avoid t.



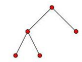
$$\rightarrow \sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k(\quad)$$



   (b) *Case* 2: We will count all of the ways to have the left child of the root be a leaf, while an $(n-1)$-leaf tree avoiding t extends from the right child of the root.

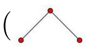
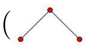
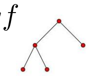
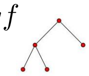


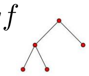
$$\rightarrow Av_{n-1}(t)$$

(c) Since there is only one way for a k-leaf tree to avoid  in case 1, we can add these cases and simplify it to

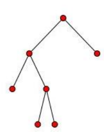
$$Av_n(t) = \sum_{k=0}^{n-2} Av_{n-k-1}(t).$$

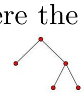
Using our Generating Function Algorithm, we have:

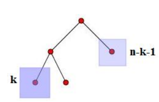
- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$
- $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot)$

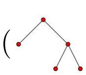
We know gf  $(\cdot) = \frac{x}{1-x}$ and $gf_t(\cdot) = x$, so by substitution:

$$\begin{aligned} gf_t(\cdot) - \frac{x}{1-x} \cdot gf_t(\cdot) &= x \\ gf_t(\cdot) \cdot (1 - \frac{x}{1-x}) &= x \\ gf_t(\cdot) &= \frac{x(1-x)}{1-2x} \\ gf_t(\cdot) &= \frac{x - x^2}{1-2x} \end{aligned}$$

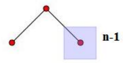
2. Let $t =$ 

(a) *Case* 1: Here the k-leaf subtree extending from the leftmost vertex must avoid , while the rightmost vertex is a $(n-k-1)$-leaf subtree that only has to avoid t, and we assume that the right vertex in the second generation must be a leaf.

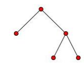


$\rightarrow \sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $)$

(b) *Case* 2: Again, we will assume the left child of the root must be a leaf. Then the possible trees extending from the right child must have $n-1$ leaves and avoid t.

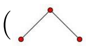
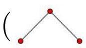
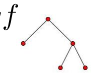
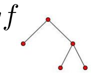


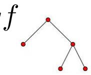
$\to Av_{n-1}(t)$

(c) There is only one way for a k-leaf tree to avoid , so the total number of n-leaf trees avoiding t simplifies to

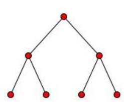
$$Av_n(t) = \sum_{k=0}^{n-2} Av_{n-k-1}(t).$$

Using our Generating Function Algorithm, we have:

- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$
- $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot)$

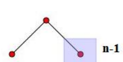
We know gf  $(\cdot) = \frac{x}{1-x}$ and $gf_t(\cdot) = x$, so by substitution:

$$\begin{aligned}
gf_t(\cdot) - \frac{x}{1-x} \cdot gf_t(\cdot) &= x \\
gf_t(\cdot) \cdot (1 - \frac{x}{1-x}) &= x \\
gf_t(\cdot) &= \frac{x(1-x)}{1-2x} \\
gf_t(\cdot) &= \frac{x-x^2}{1-2x}
\end{aligned}$$

3. Let $t =$ 

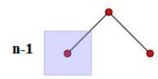
To avoid this tree, either the left child or the right child of the root may have children, but not both.

(a) *Case* 1: The left child is a leaf.
Then the right child may have any tree pattern extending from it with $n-1$ leaves which avoid t itself.



$\to Av_{n-1}(t)$

(b) *Case* 2: The right child is a leaf.
Then the left child may have any tree pattern extending from it with $n-1$ leaves which avoid t itself.

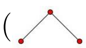
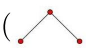
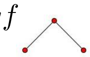
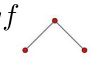
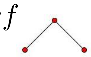
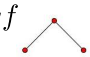


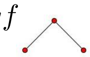
$\rightarrow Av_{n-1}(t)$

(c) We can add both of these cases to see that

$$Av_n(t) = 2Av_{n-1}(t).$$

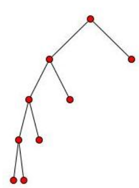
Using our Generating Function Algorithm, we have:

- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = 2gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot)^2$
- $gf_t(\cdot) = x + 2gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot)^2$

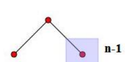
We know gf  $(\cdot) = x$ and can therefore simplify the equation to:

$$\begin{aligned} gf_t(\cdot) &= x + 2x \cdot gf_t(\cdot) - x^2 \\ gf_t(\cdot) - 2x \cdot gf_t(\cdot) &= x - x^2 \\ gf_t(\cdot)(1-2x) &= x - x^2 \\ gf_t(\cdot) &= \frac{x - x^2}{1-2x} \end{aligned}$$

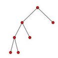
2.3 5 Leaf Trees

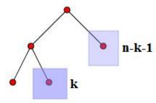
1. Let $t =$ 

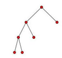
   (a) *Case* 1: Here we assume that the left subtree extending from the root is a leaf and the right subtree is an $(n-1)$-leaf tree that avoids t.

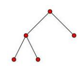
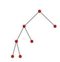
   

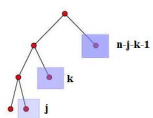
   $\rightarrow Av_{n-1}(t)$

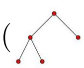
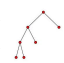
   (b) *Case* 2: Next we assume that the left subtree extending from the root is a 2-leaf tree whose left child is a leaf and whose right child is a k-leaf tree that avoids  , because this vertex is already connected to the left child of the root, and thus already a part of t. From the right child of the root, we assume there is an $(n-k-1)$-leaf tree which avoids t.



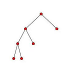
$$\rightarrow \sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k(\quad)$$ 

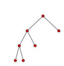
(c) *Case* 3: Then we assume that the left child of the root starts as a left leaning 3-leaf tree whose leftmost vertex is a leaf, whose middle vertex is a j-leaf tree that avoids , and whose rightmost vertex is a k-leaf tree that avoids  . The right child of the root is then an $(n-j-k-1)$-leaf tree that avoids t.



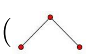
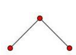
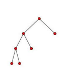
$$\rightarrow \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_j(\quad) \cdot Av_{n-j-k-1}(t) \right) Av_k(\quad)$$  

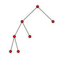
(d) We then add the above 3 cases to get the total number of tree which avoid t.

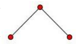
$$Av_n(t) = \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_{n-j-k-1}(t) \right) Av_k(\quad) +$$ 

$$\sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k(\quad) + Av_{n-1}(t)$$ 

Using our Generating Function Algorithm:

- $gf_t(\cdot) = x + gf_t(\quad)$ 
- $gf_t(\quad) = gf\quad(\cdot) \cdot gf_t(\cdot)$  

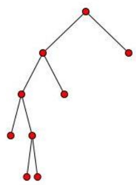
We know that gf  $= \frac{x-x^2}{1-2x}$, so by substituting:

$gf_t($  $) = \frac{x-x^2}{1-2x} \cdot gf_t(\cdot)$

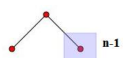
$$gf_t(\cdot) = x + \frac{x-x^2}{1-2x} \cdot gf_t(\cdot)$$

$$gf_t(\cdot) \cdot (1 - \frac{x-x^2}{1-2x}) = x$$

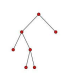
$$gf_t(\cdot) = \frac{x-2x^2}{1-3x+x^2}$$

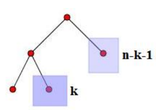
2. Let $t =$ 

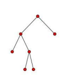
   (a) *Case* 1: Here we assume that the left child of the root is a leaf and the right child is an $(n-1)$-leaf subtree that avoids t.

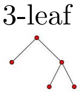
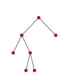
   

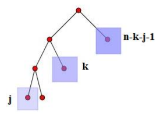
   $\rightarrow Av_{n-1}(t)$

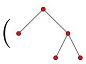
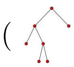
   (b) *Case* 2: Next we assume that the left subtree extending from the root is a 2-leaf tree whose left child is a leaf and whose right child is a k-leaf tree that avoids , because this vertex is already connected to the left child of the root, and thus already a part of a copy of t. From the right child of the root, we assume there is an $(n-k-1)$-leaf subtree which avoids t.

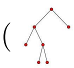
   $\rightarrow \sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $)$

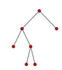
   (c) *Case* 3: Then we assume that the left subtree extending from the root starts as a 3-leaf left comb whose leftmost vertex is a j-leaf tree that avoids , whose middle vertex is a leaf, and whose rightmost vertex is a k-leaf tree that avoids . The right child of the root is then an $(n-j-k-1)$-leaf tree that avoids t.



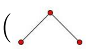
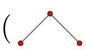
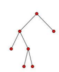
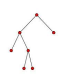
$\rightarrow \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_j($  $) \cdot Av_{n-j-k-1}(t) \right) Av_k($  $)$

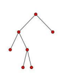
(d) We then add the above 3 cases to get the total number of tree which avoid t.

$Av_n(t) = \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_{n-j-k-1}(t) \right) Av_k($  $)+$

$\sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $) + Av_{n-1}(t)$

Using our Generating Function Algorithm:

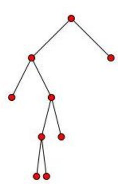
- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$
- $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot)$

We know that gf  $= \frac{x-x^2}{1-2x}$, so by substituting:

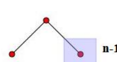
$$gf_t(\cdot) = x + \frac{x-x^2}{1-2x} \cdot gf_t(\cdot)$$

$$gf_t(\cdot) \cdot (1 - \frac{x-x^2}{1-2x}) = x$$

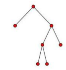
$$gf_t(\cdot) = \frac{x-2x^2}{x^2-3x+1}$$

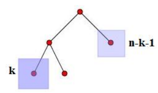
3. Let $t =$ 

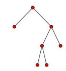
(a) *Case* 1: Here we assume that the left child of the root is a leaf and the right child of the root is an $(n-1)$-leaf tree that avoids t.

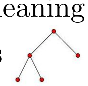
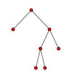


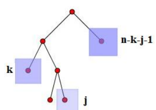
$\rightarrow Av_{n-1}(t)$

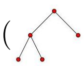
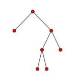
(b) *Case* 2: Next we assume that the left subtree extending from the root is a 2-leaf tree whose right child is a leaf and whose left child is a k-leaf tree that avoids , because this vertex is already connected to the root, and thus already a part of t. From the right child of the root, we assume there is an $(n-k-1)$-leaf subtree which avoids t.



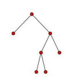
$\rightarrow \sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $)$

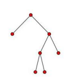
(c) *Case* 3: Then we assume that the left subtree of the root starts as a right leaning 3-leaf tree whose rightmost vertex is a j-leaf tree that avoids , whose middle vertex is a leaf, and whose leftmost vertex is a k-leaf tree that avoids . The right subtree of the root is then an $(n-j-k-1)$-leaf tree that avoids t.



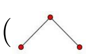
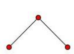
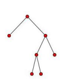
$\rightarrow \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_j($  $) \cdot Av_{n-j-k-1}(t) \right) Av_k($  $)$

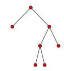
(d) We then add the above 3 cases to get the total number of trees which avoid t.

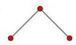
$Av_n(t) = \sum_{k=1}^{n-3} \left(\sum_{j=1}^{n-k-2} Av_{n-j-k-1}(t) \right) Av_k($  $)+$

$\sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $) + Av_{n-1}(t)$

Using our Generating Function Algorithm:

- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$

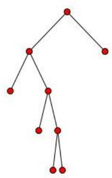
We know that gf  $= \frac{x-x^2}{1-2x}$, so by substituting:

$gf_t($  $) = \dfrac{x-x^2}{1-2x} \cdot gf_t(\cdot)$

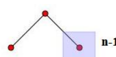
$$gf_t(\cdot) = x + \frac{x-x^2}{1-2x} \cdot gf_t(\cdot)$$

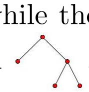
$$gf_t(\cdot) \cdot (1 - \frac{x-x^2}{1-2x}) = x$$

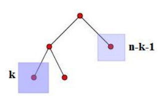
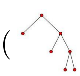
$$gf_t(\cdot) = \frac{x-2x^2}{x^2-3x+1}$$

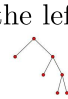
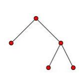
4. Let $t =$ 

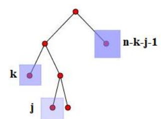
   (a) *Case* 1: First we assume the left child of the root is a leaf, thus the possible subtrees that branch from the right leaf of the root will have $n-1$ leaves and avoid t.

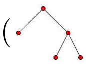
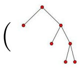
    $\rightarrow Av_{n-1}(t)$

   (b) *Case* 2: Then we assume that the right child of the second generation will be a leaf, while the k-leaf subtree extending from the leftmost vertex will avoid , and the rightmost vertex is an $n-k-1$-leaf subtree that avoids t.

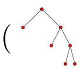
    $\rightarrow \displaystyle\sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k($  $)$

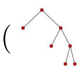
   (c) *Case* 3: Next we assume that the right child in the third generation will be a leaf, the right child of the root is an $(n-k-j-1)$-leaf tree that avoids t, the left child in the second generation is a k-leaf subtree that avoids , and the left child in the third generation is a j-leaf subtree that avoids .



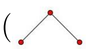
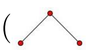
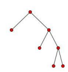
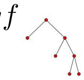
$\rightarrow \sum_{k=1}^{n-3}\left(\sum_{j=1}^{n-k-2} Av_j($  $) \cdot Av_{n-j-k-1}(t)\right) Av_k($  $)$

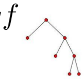
(d) We add these three cases to get:

$$Av_n(t) = \sum_{k=1}^{n-3}\left(\sum_{j=1}^{n-k-2} Av_{n-j-k-1}(t)\right) Av_k(\quad)+$$


$$\sum_{k=1}^{n-2} Av_{n-k-1}(t) \cdot Av_k(\quad) + Av_{n-1}(t)$$


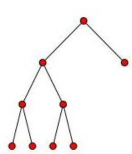
Using our Generating Function Algorithm, we have:

- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$
- $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot)$

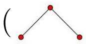
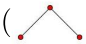
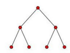
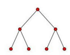
We know that gf  $(\cdot) = \frac{x-x^2}{1-2x}$, so by substituting:

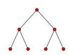
$$gf_t(\cdot) = x + \frac{x-x^2}{1-2x} \cdot gf_t(\cdot)$$
$$gf_t(\cdot) - \frac{x-x^2}{1-2x} \cdot gf_t(\cdot) = x$$
$$gf_t(\cdot)(1 - \frac{x-x^2}{1-2x}) = x$$
$$gf_t(\cdot) = \frac{x-2x^2}{x^2-3x+1}$$

Note: The summations for the remaining 5-leaf trees are complicated and from here on out we will only calculate the generating functions.

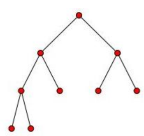
5. Let $t =$ 

Using our Generating Function Algorithm, we have:

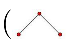
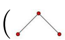
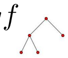
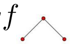
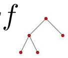
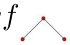
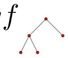
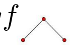
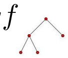
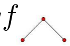
- $gf_t(\cdot) = x + gf_t($  $)$
- $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot)$
- $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot)$

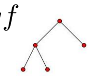
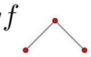
We know that gf  $(\cdot) = \frac{x-x^2}{1-2x}$, so by substituting:

$$\begin{aligned} gf_t(\cdot) &= x + \frac{x-x^2}{1-2x} \cdot gf_t(\cdot) \\ gf_t(\cdot) - \frac{x-x^2}{1-2x} \cdot gf_t(\cdot) &= x \\ gf_t(\cdot)(1 - \frac{x-x^2}{1-2x}) &= x \\ gf_t(\cdot) &= \frac{x-2x^2}{x^2-3x+1} \end{aligned}$$

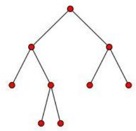
6. Let $t =$ 

   Using our Generating Function Algorithm, we have:

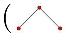
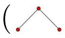
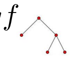
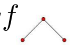
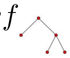
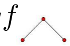
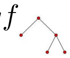
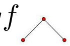
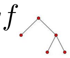
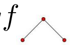
   - $gf_t(\cdot) = x + gf_t($  $)$
   - $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf$  $(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$
   - $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf$  $(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$

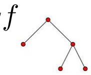
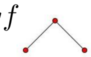
   We know that gf  $(\cdot) = \frac{x}{1-x}$ and gf  $(\cdot) = x$, so by substituting:

$$\begin{aligned} gf_t(\cdot) &= x + \frac{x}{1-x} \cdot gf_t(\cdot) + gf_t(\cdot) \cdot x - \frac{x^2}{1-x} \\ gf_t(\cdot) - \frac{x}{1-x} \cdot gf_t(\cdot) - gf_t(\cdot) \cdot x &= x - \frac{x^2}{1-x} \\ gf_t(\cdot)(1 - \frac{x}{1-x} - x) &= x - \frac{x^2}{1-x} \\ gf_t(\cdot) &= \frac{x-2x^2}{1-3x+x^2} \end{aligned}$$

7. Let $t =$ 

   Using our Generating Function Algorithm, we have:

   - $gf_t(\cdot) = x + gf_t($  $)$
   - $gf_t($  $) = gf$  $(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf$  $(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$
   - $gf_t(\cdot) = x + gf$  $(\cdot) \cdot gf_t(\cdot) + gf_t(\cdot) \cdot gf$  $(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$

   We know that gf  $(\cdot) = \frac{x-x^2}{1-2x}$ and gf  $(\cdot) = x$, so by substituting:

$$\begin{aligned}
gf_t(\cdot) &= x + \frac{x}{1-x} \cdot gf_t(\cdot) + gf_t(\cdot) \cdot x - \frac{x^2}{1-x} \\
gf_t(\cdot) - \frac{x}{1-x} \cdot gf_t(\cdot) - gf_t(\cdot) \cdot x &= x - \frac{x^2}{1-x} \\
gf_t(\cdot)(1 - \frac{x}{1-x} - x) &= x - \frac{x^2}{1-x} \\
gf_t(\cdot) &= \frac{x - 2x^2}{1 - 3x + x^2}
\end{aligned}$$

Notice how similar equation (3) is to the generating functions obtained from avoiding 3-, 4-, and 5-leaf tree patterns. At this point, we might conjecture that $gf_t(\cdot)$ depends only on the number of leaves of t. We will confirm this observation in **Theorem 2**.

3 An Interesting Generalization

Lemma 1. *Let gf_n be the generating function that counts trees avoiding an n-leaf left comb. Then $gf_k = \frac{x}{1-gf_{k-1}}$ for all $k \geq 2$.*

Proof. Let t_1 equal the n-leaf left comb and t_2 equal the $(n-1)$-leaf left comb. Using equation (3) we have,

$$gf_{t_1}(\cdot) = \frac{x - gf_{t_2}(\cdot) \cdot gf_{\cdot}(\cdot)}{1 - gf_{t_2}(\cdot) - gf_{\cdot}(\cdot)}.$$

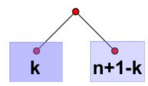
We know gf.$(\cdot) = 0$.

$$\therefore gf_{t_1}(\cdot) = \frac{x}{1-gf_{t_2}(\cdot)}.$$

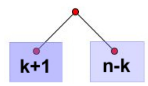
$\square$

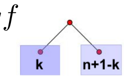
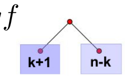
Theorem 2. *For a given $n \in \mathbb{Z}^+$, all n-leaf trees have the same avoidance generating function for non-consecutive pattern avoidance.*

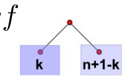
Proof.
Base Case: We have shown that this theorem is true for all $n \leq 5$.
Inductive Step: We assume true for all $k \leq n$.
We will consider two different trees with $n+1$ leaves.
First consider a tree of the form:

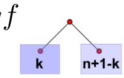


where t_ℓ is the k-leaf subtree in the left box and t_r is the $(n+1-k)$-leaf subtree in the right box.
Also consider a second $(n+1)$-leaf tree of the form



where t_ℓ is the $(k+1)$-leaf subtree in the left box and t_r is the $(n-k)$-leaf subtree in the right box.
We must show that gf  $= gf$  .

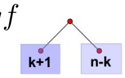
Expressing gf  using our generating function algorithm, we get:

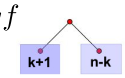
gf 

$$= \frac{x - gf_k \cdot gf_{n+1-k}}{1 - gf_k - gf_{n+1-k}}$$

$$= \frac{x - gf_k \cdot \left(\frac{x}{1-gf_{n-k}}\right)}{1 - gf_k - \left(\frac{x}{1-gf_{n-k}}\right)}$$

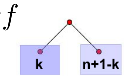
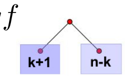
$$= \frac{x \cdot (-1 + gf_k + gf_{n-k})}{1 - gf_k - x - gf_{n-k} + gf_k \cdot gf_{n-k}}$$

Expressing gf  using our generating function algorithm, we get:

gf  $= \dfrac{x - gf_{k+1} \cdot gf_{n-k}}{1 - gf_{k+1} - gf_{n-k}}$

$$= \frac{x - gf_{n-k} \cdot \left(\frac{x}{1-gf_k}\right)}{1 - gf_{n-k} - \left(\frac{x}{1-gf_k}\right)}$$

$$= \frac{x \cdot (-1 + gf_k + gf_{n-k})}{1 - gf_k - x - gf_{n-k} + gf_k \cdot gf_{n-k}}$$

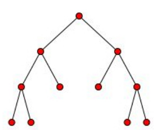
This shows that gf  $= gf$  .

$\therefore$ All n-leaf trees have the same non-consecutive avoidance generating function.

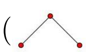
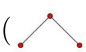
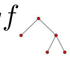
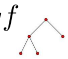
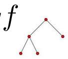
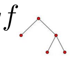
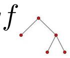
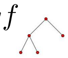
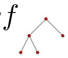
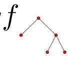
$\square$

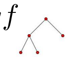
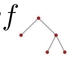
Note: From this point on, for each $n \in \mathbb{Z}^+$ we need only compute the generating function for one n-leaf tree by **Theorem 2.**

3.1 6 Leaf Trees

1. Let $t =$ 

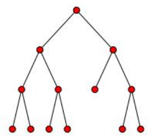
   Using our Generating Function Algorithm, we have:

   - $gf_t(\cdot) = x + gf_t($  $)$
   - $gf_t($  $) = gf_t(\cdot) \cdot gf$  $(\cdot) + gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$
   - $gf_t(\cdot) = x + gf_t(\cdot) \cdot gf$  $(\cdot) + gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$

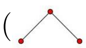
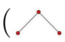
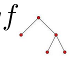
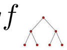
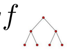
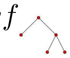
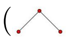
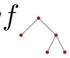
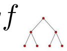
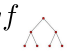
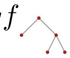
   Let $a = gf_t(\cdot)$.
   We know gf  $(\cdot) = gf$  $(\cdot) = \frac{x}{1-x}$, so by substituting:

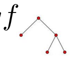
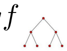
$$a = x + a\left(\frac{x}{1-x}\right) + \left(\frac{x}{1-x}\right)a - \left(\frac{x}{1-x}\right)\left(\frac{x}{1-x}\right)$$
$$a = x + \left(\frac{2ax}{1-x}\right) - \left(\frac{x^2}{(1-x)^2}\right)$$
$$gf_t(\cdot) = \frac{x - 3x^2 + x^3}{1 - 4x + 3x^2}$$

3.2 7 Leaf Trees

1. Let $t=$ 

   Using our Generating Function Algorithm, we have:

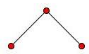
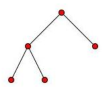
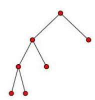
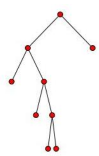
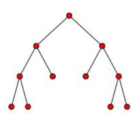
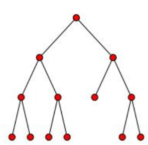
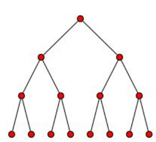
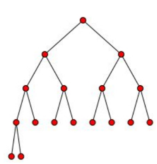
   - $gf_t(\cdot) = x + gf_t($  $)$
   - $gf_t($  $) = gf_t(\cdot) \cdot gf$  $(\cdot) + gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot) \cdot gf$  $(\cdot)$
   - $gf_t(\cdot) = x + gf_t($  $) = gf_t(\cdot) \cdot gf$  $(\cdot) + gf$  $(\cdot) \cdot gf_t(\cdot) - gf$  $(\cdot) \cdot$ gf  $(\cdot)$

   Let $gf_t(\cdot) = a$.
   We know that gf  $(\cdot) = \frac{x}{1-x}$ and gf  $(\cdot) = \frac{x-x^2}{1-2x}$, so by substituting:

$$a = x + \frac{a \cdot x}{1-x} + \frac{a \cdot (x - x^2)}{1 - 2x} - \frac{x \cdot (x - x^2)}{(1-2x) \cdot (1-x)}$$
$$a \cdot (1 - \frac{x}{1-x} - \frac{x - x^2}{1-2x}) = x - \frac{x \cdot (x - x^2)}{(1-2x) \cdot (1-x)}$$
$$a = \frac{x - \frac{x \cdot (x - x^2)}{(1-2x) \cdot (1-x)}}{1 - \frac{x}{1-x} - \frac{x - x^2}{1-2x}}$$
$$a = \frac{-3x^3 + 4x^2 - x}{x^3 - 6x^2 + 5x - 1}$$
$$gf_t(\cdot) = \frac{x - 4x^2 + 3x^3}{1 - 5x + 6x^2 - x^3}$$

4 Condensed Findings

Listed below are the generating functions, sequences, and recurrences for avoiding trees with up to 9 leaves.

t	gf_t	Sequence	Recurrence
	0	$0, 0, 0, \ldots$	$a_n = 0$
	x	$1, 0, 0, 0, \ldots$	$a_n = 0$
	$\frac{1}{1-x}$	$1, 1, 1, 1, \ldots$	$a_n = 1$
	$\frac{x-x^2}{1-2x}$	$1, 1, 2, 4, 8, 16, \ldots$	$a_n = 2 \cdot a_{n-1}$
	$\frac{x-2x^2}{1-3x+x^2}$	$1, 1, 2, 5, 13, 34, 89, \ldots$	$a_n = 3 \cdot a_{n-1} - a_{n-2}$
	$\frac{x-3x^2+x^3}{1-4x+3x^2}$	$1, 1, 2, 5, 14, 41, 122, \ldots$	$a_n = 4 \cdot a_{n-1} - 3 \cdot a_{n-2}$
	$\frac{x-4x^2+3x^3}{1-5x+6x^2-x^3}$	$1, 1, 2, 5, 14, 42, 131, \ldots$	$a_n = 5 \cdot a_{n-1} - 6 \cdot a_{n-2} + a_{n-3}$
	$\frac{x-5x^2+6x^3-x^4}{1-6x+10x^2-4x^3}$	$1, 1, 2, 5, 14, 42, 132, 428, \ldots$	$a_n = 6 \cdot a_{n-1} - 10 \cdot a_{n-2} + 4 \cdot a_{n-3}$
	$\frac{x-6x^2+10x^3-4x^4}{1-7x+15x^2-10x^3+x^4}$	$1, 1, 2, 5, 14, 42, 132, 429, 1429, \ldots$	$a_n = 7 \cdot a_{n-1} - 15 \cdot a_{n-2} + 10 \cdot a_{n-3} - a_{n-4}$

5 Recurrences

Notice that when avoiding 5-, 6-, 7-, and 8-leaf tree patterns, the absolute value of the coefficients of the polynomial in the denominator of the generating function are the same as the absolute value of the coefficients of the corresponding recurrence. For every non-consecutive avoidance generating function gf_t there exists a linear recurrence of degree d of the following form, where $c_i \in \mathbb{Z}$ and $a_n := Av_n(t)$:

$$a_n = c_1 \cdot a_{n-1} + c_2 \cdot a_{n-2} + c_3 \cdot a_{n-3} + c_4 \cdot a_{n-4} \cdots + c_d \cdot a_{n-d}$$

These recurrences give the sequences of the number of n-leaf trees that avoid a given tree pattern t. Let M be the following matrix, whose row entries are the coefficients of the recurrences.

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 2 & 0 & 0 & 0 & 0 & \dots & 0 \\ 3 & -1 & 0 & 0 & 0 & \dots & 0 \\ 4 & -3 & 0 & 0 & 0 & \dots & 0 \\ 5 & -6 & 1 & 0 & 0 & \dots & 0 \\ 6 & -10 & 4 & 0 & 0 & \dots & 0 \\ 7 & -15 & 10 & -1 & 0 & \dots & 0 \\ 8 & -21 & 20 & -5 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{n-2}{1} & -\binom{n-3}{2} & \binom{n-4}{3} & -\binom{n-5}{4} & \binom{n-6}{5} & \dots & -1^k \cdot \binom{n-k}{k-1} \end{bmatrix}$$

Note: Starting with the first non-zero entry in every column, the columns form the diagonals of Pascal's triangle. Following this pattern, we can deduce the recurrence, and thus the sequence, for trees that avoid any tree pattern non-consecutively.

5.1 Generalized Generating Function

We were able to extend the pattern in the recurrences to the corresponding avoidance generating functions. We then discovered that all generating functions for non-consecutive tree avoidance can be written as rational polynomials whose coefficients are binomial coefficients.

Theorem 3. *For a given $n \in \mathbb{Z}^+$, the non-consecutive avoidance generating function for any n-leaf tree is given by*

$$gf_n = \frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{n-(k+2)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{n-(k+1)}{k} \cdot x^k}.$$

Proof.
<u>Base Case</u>: We have shown that this theorem is true for all $n \leq 9$.
<u>Inductive Step</u>: We assume it is true for all $k \leq n$.
We want to show that this holds for gf_{n+1}, *i.e.* that:

$$gf_{n+1} = \frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+2)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+1)}{k} \cdot x^k} \tag{4}$$

From **Lemma 1**
$$gf_{n+1} = \frac{x}{1 - gf_n}.$$

Consider the right hand side of equation (4).
We wish to show that

$$\frac{x}{1 - gf_n} = \frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+2)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+1)}{k} \cdot x^k}.$$

On the righthand side,

$$\frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+2)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+1)}{k} \cdot x^k} = \frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n)-(k+1)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+1)}{k} \cdot x^k}$$

because $(n+1) - (k+2) = n - (k+1)$.
Then on the lefthand side,

$$\frac{x}{1-gf_n} = \frac{x}{1-\left(\frac{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+2)}{k}\cdot x^{k+1}}{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k}\right)}.$$
$$= \frac{x}{\frac{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k-\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+2)}{k}\cdot x^{k+1}}{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k}}.$$
$$= \frac{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^{k+1}}{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k-\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+2)}{k}\cdot x^{k+1}}.$$

We now wish to show that,

$$\frac{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^{k+1}}{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k-\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+2)}{k}\cdot x^{k+1}} = \frac{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{(n)-(k+1)}{k}\cdot x^{k+1}}{\sum_{k=0}^{\infty}(-1)^k\cdot\binom{(n+1)-(k+1)}{k}\cdot x^k}$$

We can see here that the numerators of the two fractions are equal. From this point on we will only look at the denominators. Looking at the denominator on the left side of the equation, we can pull out the $k=0$ term in the first sum, and also let $k+1=j$ in the second sum.

$$\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+1)}{k}\cdot x^k-\sum_{k=0}^{\infty}(-1)^k\cdot\binom{n-(k+2)}{k}\cdot x^{k+1}$$

$$= 1+\sum_{j=1}^{\infty}(-1)^j\cdot\binom{n-(j+1)}{j}\cdot x^j-\sum_{j=1}^{\infty}(-1)^{j-1}\cdot\binom{n-(j+1)}{j-1}\cdot x^j$$
$$= 1+\sum_{j=1}^{\infty}(-1)^j\cdot\left(\binom{n-(j+1)}{j}+\binom{n-(j+1)}{j-1}\right)\cdot x^j$$
$$= 1+\sum_{j=1}^{\infty}(-1)^j\cdot\binom{(n+1)-(j+1)}{j}\cdot x^j$$
$$= \sum_{k=0}^{\infty}(-1)^k\cdot\binom{(n+1)-(k+1)}{k}\cdot x^k\text{, as desired.}$$

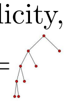
$\therefore$

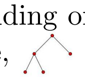
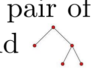
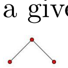
$$\frac{x}{1-gf_n} = \frac{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+2)}{k} \cdot x^{k+1}}{\sum\limits_{k=0}^{\infty} (-1)^k \cdot \binom{(n+1)-(k+1)}{k} \cdot x^k}.$$

We have proven that this theorem holds for any $n+1$. Thus by induction, **Theorem 3** is true. $\square$

Thus far we have proven that the generating functions for non-consecutive avoidance are rational and that the corresponding sequences satisfy linear recurrences. Next, we will further examine these recurrences by trying to find a structural explanation for them within the trees themselves.

5.2 Explaining the 5-leaf Recurrence

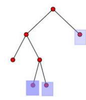
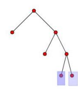
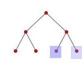
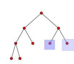
We have shown that the number of n-leaf trees that avoid any given 5-leaf tree pattern non-consecutively can be represented by the recurrence $a_n = 3 \cdot a_{n-1} - a_{n-2}$, where $a_1 = a_2 = 1$. Note that the given recurrence is equivalent to $a_n = 3 \cdot (a_{n-1} - a_{n-2}) + 2 \cdot a_{n-2}$, where $a_1 = a_2 = 1$. Our goal is to show how this recurrence supports the construction of every n-leaf tree that avoids a given 5-leaf tree pattern. For simplicity, we have chosen to demonstrate this explanation with trees that avoid $t =$  .

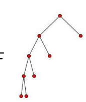
We want to be able to construct every n-leaf tree that avoids this left-leaning 5-leaf comb from all of the $(n-1)$-leaf trees. To do this, we must first define the new terms we will use in our explanation. A *descendant* refers to any m-leaf tree that is constructed by adding one pair of children to one leaf of a given $(m-1)$-leaf tree. For example,  and  are the descendants of  because they are constructed by adding exactly two children to each leaf of the 2-leaf tree.

We have developed a set of rules that show how to construct all n-leaf trees that avoid t from the $(n-1)$-leaf trees. Remember that $a_n = 3 \cdot (a_{n-1} - a_{n-2}) + 2 \cdot a_{n-2}$. Notice that the coefficients in this formula are 2 and 3. We will create either 2 or 3 descendants for every $(n-1)$-leaf tree to structurally explain the recurrence. It is obvious that for any $(n-1)$-leaf tree, there are $n-1$ possible descendants, but if we constructed all of these descendants, there would be many trees that are constructed multiple times. Thus, we have to decide which leaves of each $(n-1)$-leaf tree will have children and which will remain leaves.

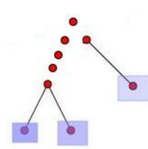
The first rule in our process of deciding which leaves have children is that we must avoid t when creating descendants. For example, the leftmost leaf of  cannot have children because it would create t.

The second rule is that a leaf will receive children unless it is possible to backtrack towards the root and come to a vertex whose right child is an internal vertex not on the initial path traveled. The examples below follow this rule:

- The leftmost child of  does not have a descendant because the right child in its same generation has children already.
- The only descendants of  are from the children of the third generation because the right children of the first and second generations have children already, and so the left children of the first and second generations cannot have descendants.
- The two left children of the second generation of  cannot have descendants because the right child in the first generation has children already.
- Finally, in , the children of the third generation and the leftmost leaf of the second generation cannot have descendants because the first generation has children on the right.

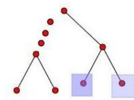
Theorem 4. *A given n-leaf tree that avoids the tree pattern t=*  *non-consecutively has either 2 or 3 descendants.*

Proof.
Base Case: We have shown that this theorem is true for all $n \leq 3$.
Inductive Step: We assume the theorem is true for all $k \leq n$.
Assume we have a k-leaf tree that has three options in which to expand. A form of this tree will be



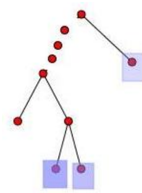
We will consider the three different cases with $k+1$ leaves:

- *Case* 1: Add children to the right vertex



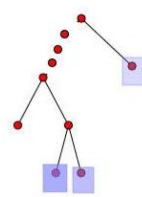
According to the rules defined earlier, we can now only add children from the two rightmost vertices. Thus this $(k+1)$-leaf tree has two descendants.

- *Case* 2: Add children to the middle vertex



Again, the rules allow us to add children to all the vertices except for the leftmost one. Thus this $(k+1)$-leaf tree has three descendants.

- *Case* 3: Add children to the left vertex

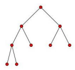
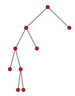
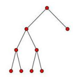


Using our rules, we can deduce that this $(k+1)$-leaf tree has three descendants. We cannot add children to the leftmost leaf due to the fact that it will not avoid t.

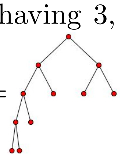
By adding to the rightmost possible leaf, we eliminate the other two possible leaves as choices to add. This was shown in *Case 1.* When considering all possible k-leaf trees, we know that each $(k-1)$-leaf tree will have two or three choices. However, if a specific $(k-1)$-leaf tree was created recursively by adding to the rightmost choice of a $(k-2)$-leaf tree, then we see that it will have only 2 possible leaves that can have children. If the $(k-1)$-leaf tree was created by adding to the right of a $(k-2)$-leaf tree, it will account for two k-leaf trees. Thus, each of the $(k-2)$-leaf trees will account for a $(k-1)$-leaf tree with only two possible choices, and the rest of the $(k-1)$-leaf trees will have three choices.

∴ We have proven that any $(k+1)$-leaf tree will have either two or three descendants. Thus by induction, the given theorem is true. $\square$

Note: Through similar reasoning, this theorem is true for:

$t =$ , $t =$ , and $t =$ .

5.3 Conjecture for Explaining the 6-Leaf Recurrence and Beyond

Having examined the structure behind the 5-leaf recurrence in depth, we decided to further investigate the structure behind the recurrence for avoiding 6-leaf trees. In order to avoid the 6-leaf left comb, we started with all the 5-leaf trees and created 6-leaf trees by following the same rules we created for avoiding a 5-leaf tree. In this manner, we arrived at 41 6-leaf trees that avoid the 6-leaf left comb, which is the correct number. Each 5-leaf tree that we started with had either 4, 3, or 2 descendants, with four trees having 4, five trees having 3, and five trees having 2. We repeated this process for avoiding $t=$  and arrived at the same result. Because our rules seem to be working for the 5-leaf and 6-leaf trees, we think it could work for the other recurrences, but we were unable to fit the pattern of descendants from the 5-leaf trees to the recurrence for avoiding 6-leaf trees. However, another pattern seems to be emerging in the number of possible descendants when following our rules. When trying to avoid a 5-leaf tree, the trees we start with have either 3 or 2 descendants, and when trying to avoid a 6-leaf tree, the trees we start with have 4, 3, or 2 descendants. Thus, we conjecture that this pattern will continue for any n. So, when avoiding an n-leaf tree, the trees we start with will have $(n-2)$, $(n-3)$, $\ldots$, or 2 descendants. Examining this conjecture is a topic for further study. Now that we have examined how the n-leaf recurrences arise from the structure of trees, we will now further study the recurrences by examining their characteristic polynomials.

5.4 Characteristic Polynomials of Recurrences

Because we have rational generating functions, we know that the sequences they encode satisfy linear recurrences with constant coefficients. We know

Table 1: Summary of the Characteristic Polynomials

Number of Leaves	Recurrence	Characteristic Polynomial	Largest Root
1	$a_n = 0$	- -	- -
2	$a_n = 0$	- -	- -
3	$a_n = 1$	$r - 1 = 0$	1
4	$a_n = 2a_{n-1}$	$r - 2 = 0$	2
5	$a_n = 3a_{n-1} - a_{n-2}$	$r^2 - 3r + 1 = 0$	$\frac{3+\sqrt{5}}{2} \approx 2.618$
6	$a_n = 4a_{n-2} - 3a_{n-1}$	$r^2 - 4r + 3 = 0$	3
7	$a_n = 5a_{n-1} - 6a_{n-2} + a_{n-3}$	$r^3 - 5r^2 + 6r - 1 = 0$	≈ 3.247
8	$a_n = 6a_{n-1} - 10a_{n-2} + 4a_{n-3}$	$r^3 - 6r^2 + 10r - 4 = 0$	$2 + \sqrt{2} \approx 3.414$
9	$a_n = 7a_{n-1} - 15a_{n-2} + 10a_{n-3} - a_{n-4}$	$r^4 - 7r^3 + 15r^2 - 10r + 1 = 0$	≈ 3.532
10	$a_n = 8a_{n-1} - 21a_{n-2} + 20a_{n-3} - 5a_{n-4}$	$r^4 - 8r^3 + 21r^2 - 20r + 5 = 0$	$\frac{5+\sqrt{5}}{2} \approx 3.618$

that each of these recurrences of degree d has the form $a_n = c_1 \cdot a_{n-1} + c_2 \cdot a_{n-2} + c_3 \cdot a_{n-3} + c_4 \cdot a_{n-4} + \cdots + c_d \cdot a_{n-d}$. We can rewrite this as $a_n - c_1 \cdot a_{n-1} - c_2 \cdot a_{n-2} - \cdots - c_d \cdot a_{n-d} = 0.$. It is from the second equation that we arrive at the *characteristic polynomial* for any recurrence:

$$r^d - c_1 \cdot r^{d-1} - c_2 \cdot r^{d-2} - \cdots - c_d \cdot r^0 = 0. \tag{5}$$

The roots of this polynomial help explain how the sequence behaves asymptotically. The roots of the polynomials alternate between real and complex, but the imaginary part of the complex roots is small enough ($< 10^{-10}$) to be ignored. Additionally, all of the roots for each characteristic polynomial are positive and distinct. This means that for c, the largest root of the polynomial, $Av_n(t) \sim c^n$.

It appears that the largest root for the characteristic polynomial of the recurrences is approaching 4. Thus, $Av_n(t)$ will never grow faster than 4^n. Furthermore, another pattern emerges in factoring the characteristic polynomials.

If the number of leaves is prime, the characteristic polynomial for the recurrence satisfied by $Av_n(t)$ is not factorable. If the number of leaves is composite, however, the characteristic polynomial is factorable. If the prime factorization of n has m factors, then the characteristic polynomial for the recurrence of that n is factorable, and its factors will be at least the characteristic polynomials for all ways to multiply $m-1$ of n's prime factors. For example, the prime factorization of 12 is $2^2 \cdot 3$, so the characteristic polynomial for avoiding 12-leaf trees will factor into at least

Table 2: Factoring the Characteristic Polynomials

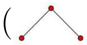
Number of Leaves	Characteristic Polynomial	Factored Characteristic Polynomial
3	$r-1$	- -
4	$r-2$	- -
5	r^2-3r+1	- -
6	r^2-4r+3	$(r-1)(r-3)$
7	r^3-5r^2+6r-1	- -
8	$r^3-6r^2+10r-4$	$(r-2)(r^2-4r+2)$
9	$r^4-7r^3+15r^2-10r+1$	$(r-1)(r^3-6r^2+9r-1)$
10	$r^4-8r^3+21r^2-20r+5$	$(r^2-3r+1)(r^2-5r+5)$
11	$r^5-9r^4+28r^3-35r^2+15r-1$	- -
12	$r^5-10r^4+36r^3-56r^2+35r-6$	$(r^2-4r+3)(r-2)(r^2-4r+1)$
13	$r^6-11r^5+45r^4-84r^3+70r^2-21r+1$	- -
14	$r^6-12r^5+55r^4-120r^3+126r^2-56r+7$	$(r^3-5r^2+6r-1)(r^3-7r^2+14r-7)$
15	$r^7-13r^6+66r^5-165r^4+210r^3-126r^2+28r-1$	$(r-1)(r^2-3r+1)(r^4-9r^3+26r^2-24r+1)$
16	$r^7-14r^6+78r^5-220r^4+330r^3-252r^2+84r-8$	$(r^3-6r^2+10r-4)(r^4-8r^3+20r^2-16r+2)$

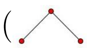
the characteristic polynomials for 4 leaves $(2 \cdot 2)$ and for 6 leaves $(2 \cdot 3)$. Additionally, the characteristic polynomial for the recurrence for avoiding 15-leaf trees will factor into at least the characteristic polynomials for 3 and 5 leaves because 3 and 5 are the only prime factors of 15. Finally, the characteristic polynomial for 16 leaves factors into the characteristic polynomial for 8 leaves multiplied by one other polynomial.

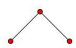
Thus far, we have only considered the enumeration of trees avoiding a single tree pattern. From this point on, we will investigate the avoidance of two tree patterns.

6 Avoiding Two n-Leaf Trees Non-Consecutively

We have discovered how to find the generating functions for non-consecutive tree avoidance and how to structurally explain the recurrences that arise from these generating functions. We will now consider avoiding two n-leaf tree patterns non-consecutively. Using our generating function algorithm for avoiding a single tree pattern as a model, we created a new algorithm for avoiding two n-leaf tree patterns simultaneously. With this new algorithm comes new notation; let $gf_{t_1,t_2}(p)$ be the generating function for the number of n-leaf binary trees that avoid the tree patterns t_1 and t_2 non-consecutively and contain the consecutive tree pattern p at their root. Because all binary trees begin with a single vertex, it follows that the generating function for all trees avoiding t_1 and t_2 is given by $gf_{t_1,t_2}(\cdot)$. Similarly, let t_{1_ℓ}, t_{2_ℓ}, t_{1_r}, t_{2_r} denote the trees descending from the the left and right children of the root of t_1 and t_2 respectively.

Since we are working with full binary trees, the root has either zero or two children. When there are zero children, we have a 1-leaf tree, which can be represented with the generating function x. When there are two children, we have a tree with two or more leaves which can be denoted by the generating function $gf_{t_1,t_2}($  $)$ which counts the number of trees that avoid t_1 and t_2, where the root has two children. Thus, our equations are as follows:

$gf_{t_1,t_2}(\cdot) = x + gf_{t_1,t_2}($  $)$ (6)

$gf_{t_1,t_2}($  $)$

$$\begin{aligned}
= {} & gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_1,t_2}(\cdot) + gf_{t_1,t_2}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot) \\
& + gf_{t_{1_\ell},t_2}(\cdot) \cdot gf_{t_1,t_{2_r}}(\cdot) + gf_{t_1,t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_2}(\cdot) \\
& - gf_{t_1,t_{2_r}}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot) - gf_{t_{1_\ell},t_2}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot) \\
& - gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_1,t_{2_r}}(\cdot) - gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_2}(\cdot) \\
& + gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot)
\end{aligned} \tag{7}$$

To clarify equation (7), we will give a term-by-term explanation.

- Assume the tree extending from the left child of the root avoids t_{1_ℓ} and t_{2_ℓ}. This means that there cannot be a copy of t_1 or t_2 that includes the root or that is to the left of the root so the right subtree need only avoid t_1 and t_2. This accounts for the term $gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_1,t_2}(\cdot)$.
- Assume the tree extending from the right child of the root avoids t_{1_r} and t_{2_r}. This means that there cannot be a copy of t_1 or t_2 that includes the root or that is to the right of the root so the left subtree need only avoid t_1 and t_2. This accounts for the term $gf_{t_1,t_2}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot)$.
- Assume that the left subtree extending from the root avoids t_{1_ℓ}. This means that the right subtree must only avoid t_1. Also, assume that the right subtree extending from the root avoids t_{2_r}. Thus, the left subtree must avoid t_2. The generating function representation of this is $gf_{t_{1_\ell},t_2}(\cdot) \cdot gf_{t_1,t_{2_r}}(\cdot)$.

- Assume that the right subtree extending from the root avoids t_{1_r}. As a result, the left subtree must only avoid t_1. Also, assume that the left subtree extending from the root avoids t_{2_ℓ}. It follows that the left subtree needs to only avoid t_2. Thus, the generating function representation of this is $gf_{t_1,t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_2}(\cdot)$.

Thus, we add these four terms in order to count the number of n-leaf trees that avoid two trees. However, this summation overcounts several cases. Therefore we need to subtract out the trees that were counted more than once.

- We subtract $gf_{t_1,t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot)$ because the trees that avoid t_1 and t_{2_l} on the left and t_{1_r} and t_{2_r} on the right are counted in both the second and fourth terms of the equation.
- We subtract $gf_{t_{1_\ell},t_2}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot)$ because the trees that avoid t_{1_ℓ} and t_2 on the left and t_{1_r} and t_{2_r} on the right are counted in both the second and third terms of the equation.
- We subtract $gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_1,t_{2_r}}(\cdot)$ because the trees that avoid t_{1_ℓ} and t_{2_ℓ} on the left and t_1 and t_{2_r} on the right are counted in both the first and third terms of the equation.
- We subtract $gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_2}(\cdot)$ because the trees that avoid t_{1_ℓ} and t_{2_ℓ} on the left and t_{1_r} and t_2 on the right are counted in both the first and fourth terms of the equation.

Finally, we must add in $gf_{t_{1_\ell},t_{2_\ell}}(\cdot) \cdot gf_{t_{1_r},t_{2_r}}(\cdot)$ because all of the trees that avoid t_{1_ℓ} and t_{2_ℓ} on the left and t_{1_r} and t_{2_r} on the right are counted four times in the first four terms and then subtracted four times in the second four terms, and thus they must be added back in at the end.

6.1 Classes of Trees

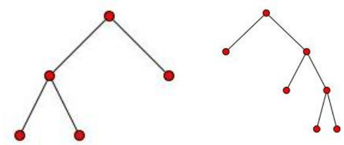
Now that we have a way to compute avoidance generating functions for two trees, we can group equivalent classes of trees together based on these generating functions.

6.1.1 Avoiding a 3-Leaf & a 4-Leaf Tree

Class A

- $gf_{t_1,t_2}(\cdot) = x + x^2 + x^3$

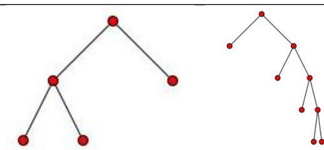
- Seq: 1, 1, 1, 0, 0, ...
- Terminates



6.1.2 Avoiding a 3-Leaf & a 5-Leaf Tree

Class A

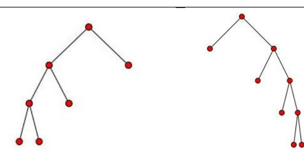
- $gf_{t_1,t_2}(\cdot) = x + x^2 + x^3 + x^4$
- Seq: 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, ...
- Terminates



6.1.3 Avoiding a 4-Leaf & a 5-Leaf Tree

Class A

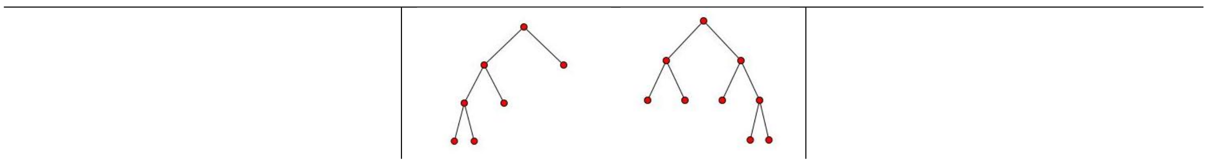
- $gf_{t_1,t_2}(\cdot) = x + x^2 + 2x^3 + 4x^4 + 7x^5 + 8x^6 + 8x^7 + 6x^8 + 3x^9 + x^{10}$
- Seq: 1, 1, 2, 4, 7, 8, 6, 3, 1, 0, 0, 0, ...
- Terminates



Class B

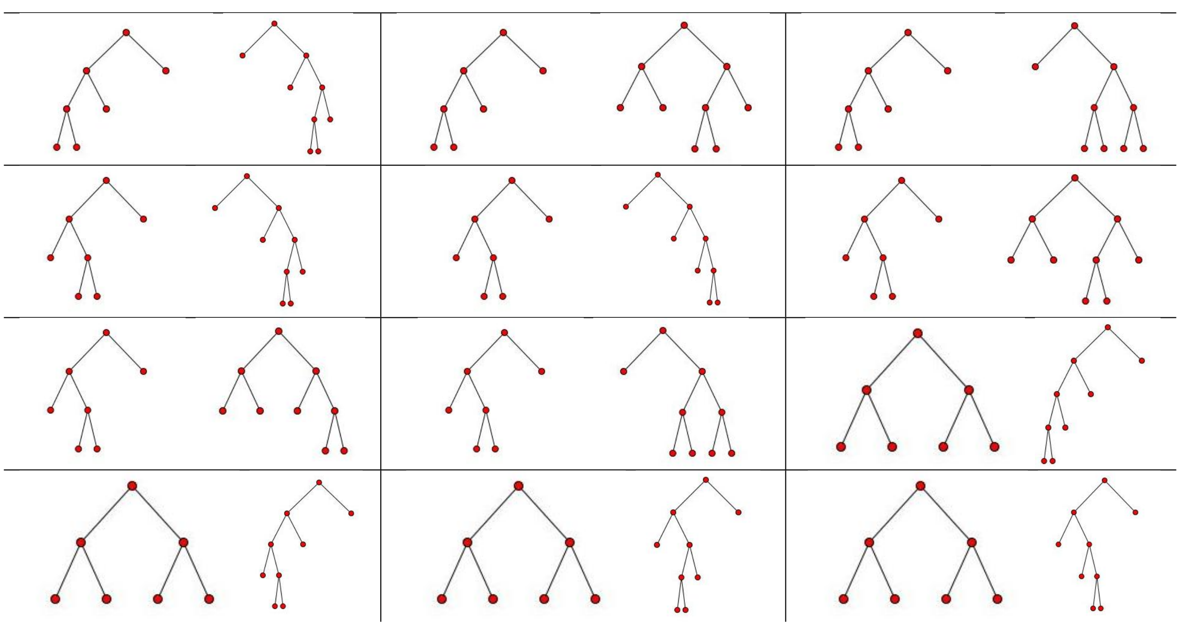
- $gf_{t_1,t_2}(\cdot) = \frac{x-x^2+x^3+x^4+x^5}{1-2x+x^2}$

- Seq: 1, 1, 2, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, ...
- OEIS A016777: $3k + 1$ for $k \geq 4$.



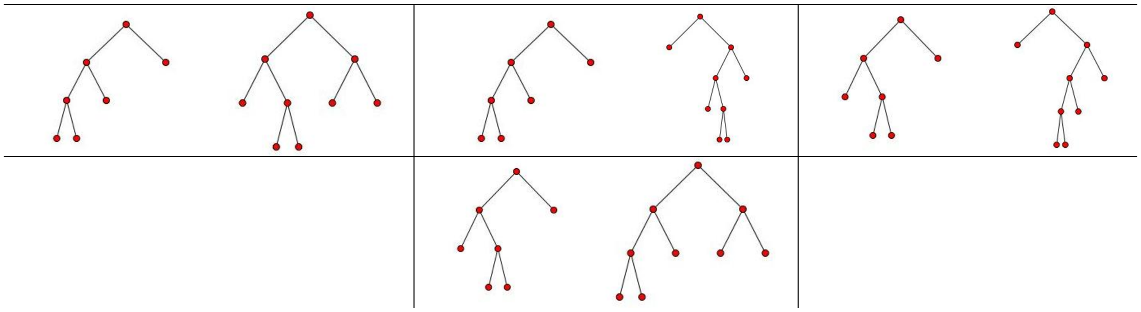
Class C

- $gf_{t_1,t_2}(\cdot) = \frac{-x+2x^2-2x^3}{-1+3x-3x^2+x^3}$
- Seq: 1, 1, 2, 4, 7, 11, 16, 22, 29, 37, 46, 56, 67, 79, 92, ...
- OEIS A152947: $\frac{(k-2)\cdot(k-1)+1}{2}$



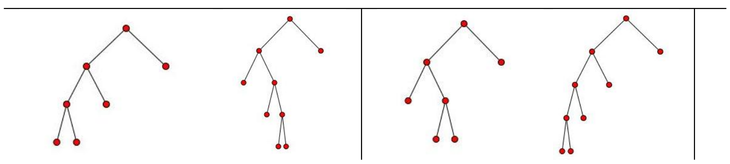
Class D

- $gf_{t_1,t_2}(\cdot) = \frac{x-x^2+x^4}{1-2x+x^3}$
- Seq: 1, 1, 2, 4, 7, 12, 20, 33, 54, 88, 143, 232, 376, 609, 986, ...
- OEIS A000071: Fibonacci numbers -1 for $n \geq 2$.



Class E

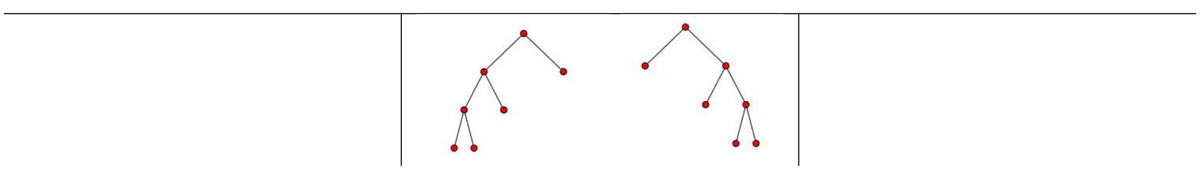
- $gf_{t_1,t_2}(\cdot) = \frac{-x}{-1+x+x^2+x^3}$
- Seq: 1, 1, 2, 4, 7, 13, 24, 44, 81, 149, 274, 504, 927, 1705, 3136, ...
- OEIS A000073: Tribonacci Numbers



6.1.4 Avoiding a Pair of 4 Leaf Trees

Class A

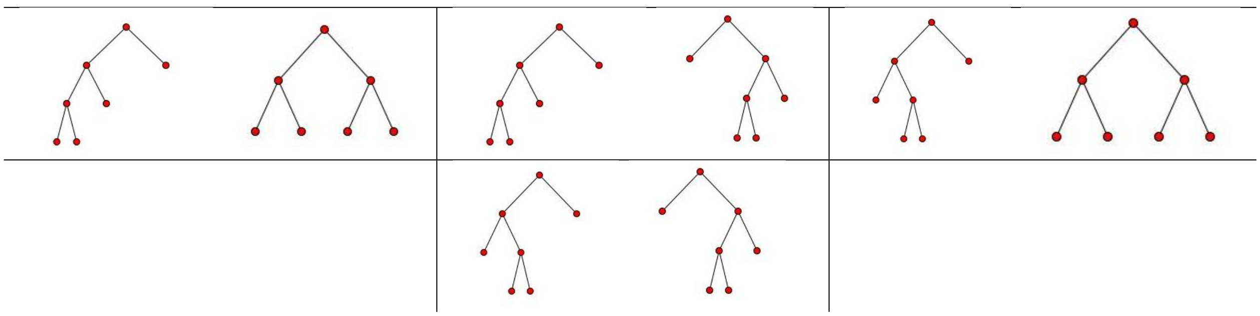
- $gf_{t_1,t_2}(\cdot) = x + x^2 + 2x^3 + 3x^4 + 2x^5 + x^6$
- Seq: 1, 1, 2, 3, 2, 1, 0, 0, 0, ...
- Terminates



Class B

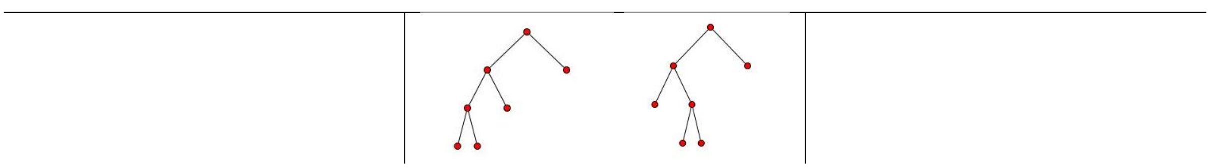
- $gf_{t_1,t_2}(\cdot) = \frac{x-x^2+x^3}{1-2x+x^2}$

- Seq: 1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, ...
- OEIS A028310: Expansion of $\frac{1-x+x^2}{(1-x)^2}$ in powers of x.



Class C

- $gf_{t_1,t_2}(\cdot) = \frac{-x}{-1+x+x^2}$
- Seq: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, ...
- OEIS A000045: Fibonacci Numbers



6.1.5 Avoiding a Pair of 5 Leaf Trees

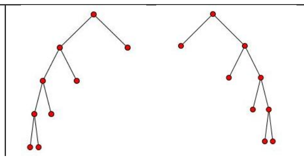
Note: The first five terms of the sequences in this section will be 1, 1, 2, 5, 12. Therefore, the sequences will begin with the sixth term.

Class A

- $\mathrm{gf}_{t_1,t_2}(\cdot) = x + x^2 + 2x^3 + 5x^4 + 12x^5 + 26x^6 + 46x^7 + 76x^8 + 116x^9 + 163x^{10} + 208x^{11} + 238x^{12} + 240x^{13} + 210x^{14} + 158x^{15} + 100x^{15} + 52x^{17} + 21x^{18} + 6x^{19} + x^{20}$

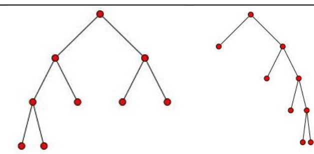
- Seq: 26, 46, 76, 116, 163, 208, 238, 240, 210, 158, 100, 52, 21, 6 , 1

- Terminates



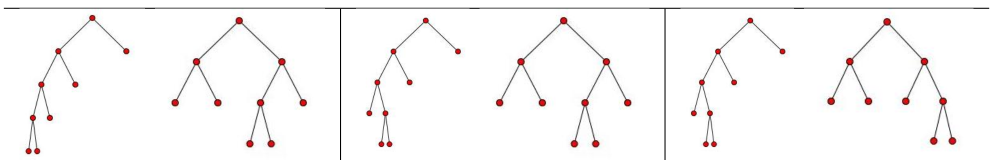
Class B

- $gf_{t_1,t_2}(\cdot) = \frac{x-2x^2+2x^3+x^4+2x^5+3x^6+2x^7+2x^8+x^9}{1-3x+3x^2-x^3}$
- Seq: 26, 49, 83, 129, 187, 257, 339, 433, 539, 657, ...
- New to OEIS



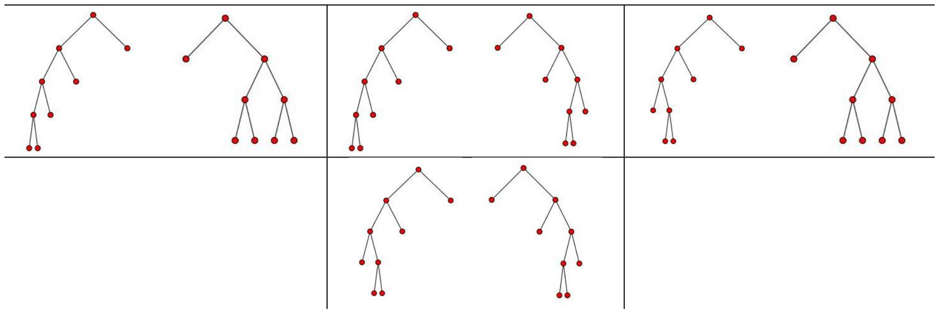
Class C

- $gf_{t_1,t_2}(\cdot) = \frac{x-4x^2+7x^3-5x^4+2x^5}{1-5x+10x^2-10x^3+5x^4-x^5}$
- Seq: 26, 51, 92, 155, 247, 376, 551, 782, 1080, 1457, ...
- OEIS A027927: $T(k, 2k-4), T$ given by A027926 for $n \geq 2$.



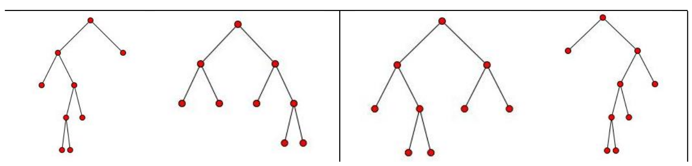
Class D

- $gf_{t_1,t_2}(\cdot) = \frac{x-5x^2+11x^3-12x^4+7x^5-2x^6+x^7}{1-6x+15x^2-20x^3+15x^4-6x^5+x^6}$
- Seq: 26, 52, 98, 176, 303, 502, 803, 1244, 1872, 2744, ...
- New to OEIS



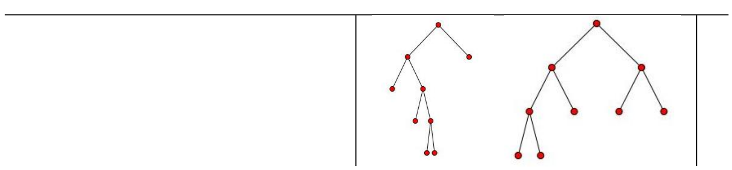
Class E

- $gf_{t_1,t_2}(\cdot) = \frac{x-3x^2+3x^3+x^4-x^5}{1-4x+5x^2-x^3-2x^4+x^5}$
- Seq: 26, 52, 98, 177, 310, 531, 895, 1491, 2463, 4044, ...
- OEIS A116717: Number of permutations of length k which avoid the patterns 231, 1423, 3214 for $n \geq 2$.



Class F

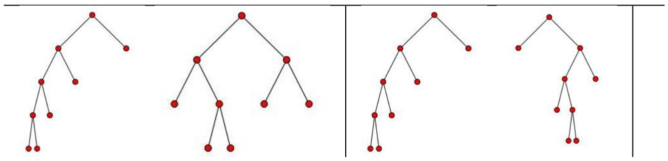
- $gf_{t_1,t_2}(\cdot) = \frac{-x+2x^2-x^3-x^4-2x^5}{-1+3x-2x^2-x^4+x^5}$
- Seq: 26, 53, 104, 199, 375, 700, 1299, 2402, 4432, 8167, ...
- New to OEIS



Class G

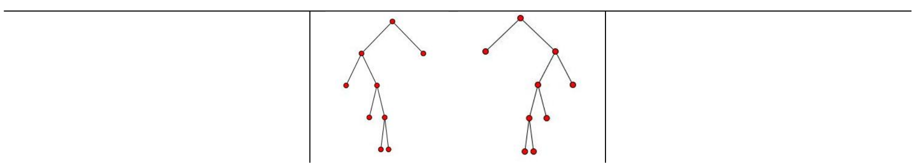
- $gf_{t_1,t_2}(\cdot) = \frac{x-2x^2+2x^4+2x^5-x^6-x^7}{1-3x+x^2+2x^3+x^4-x^5-x^6}$
- Seq: 26, 55, 113, 227, 449, 877, 1696, 3254, 6203, 11762, ...

- OEIS A116726: Number of permutations of length k which avoid the patterns 213, 1234, 2431 for $n \geq 2$.



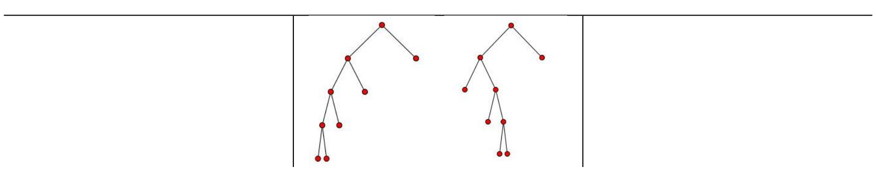
Class H

- $gf_{t_1,t_2}(\cdot) = \frac{x-x^2-x^3+3x^5+2x^6+x^7}{1-2x-x^2+3x^4+2x^5+x^6}$
- Seq: 26, 56, 118, 244, 499, 1010, 2027, 4040, 8004, 15776, ...
- OEIS A073778: $a(m) = \sum_{k=0}^{m} T(k) \cdot T(m-k)$. Convolution of tribonacci sequence A000073 with itself for $m \geq 3$, for $n \geq 2$.



Class I

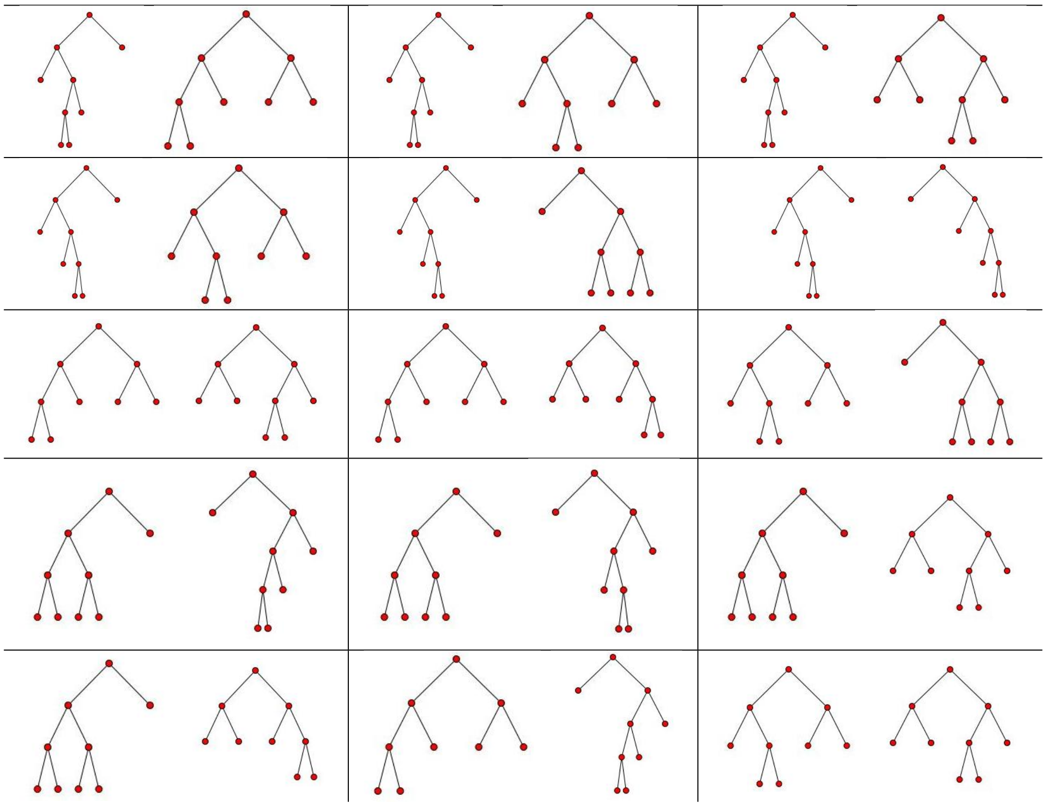
- $gf_{t_1,t_2}(\cdot) = \frac{-x}{-1+x+x^2+2x^3+3x^4+2x^5+x^6}$
- Seq: 26, 57, 127, 284, 632, 1405, 3126, 6958, 15485, 34458, ...
- New to OEIS



Class J

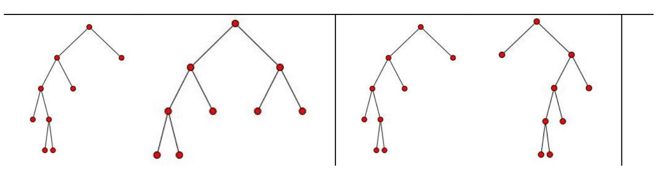
- $gf_{t_1,t_2}(\cdot) = \frac{-x+3x^2-3x^3}{-1+4x-5x^2+2x^3}$
- Seq: 27, 58, 121, 248, 503, 1014, 2037, 4084, 8179, 16370, ...

- OEIS A000325: $2^k - k$



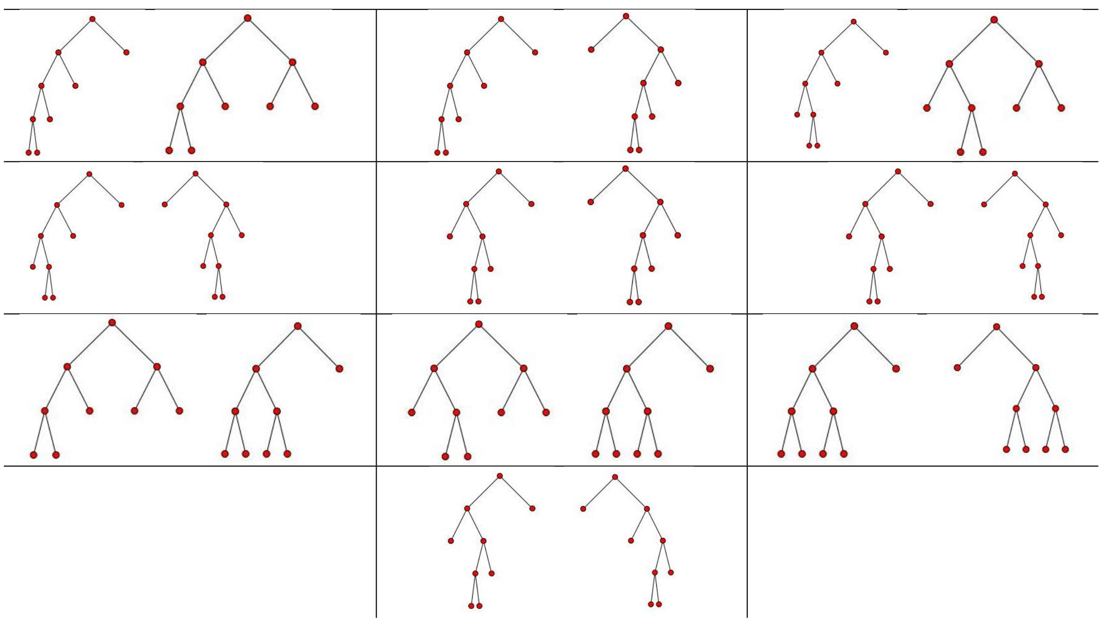
Class K

- $gf_{t_1,t_2}(\cdot) = \frac{x-2x^2+2x^4+x^5}{1-3x+x^2+2x^3}$
- Seq: 27, 59, 126, 263, 551, 1136, 2327, 4743, 9630, 19493, ...
- OEIS A116712: Number of permutations of length k which avoid the patterns 231, 3214, 4312 for $n \geq 2$.



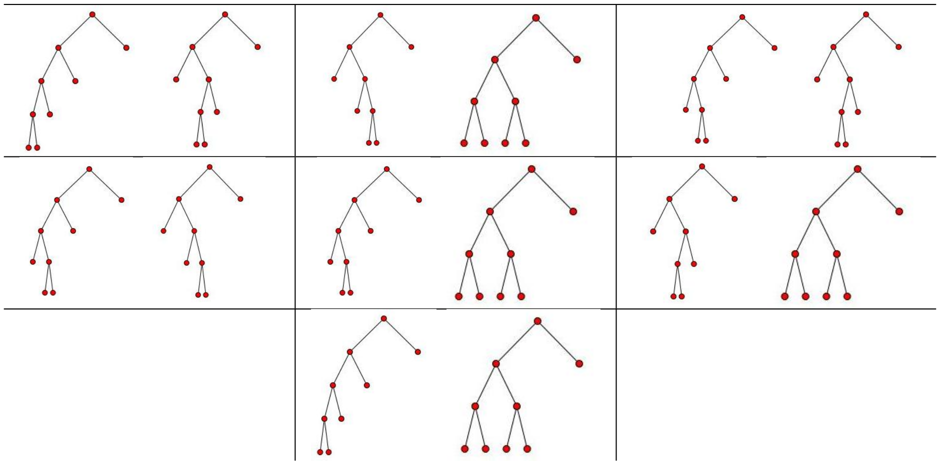
Class L

- $gf_{t_1,t_2}(\cdot) = \frac{x-3x^2+2x^3+x^4}{1-4x+4x^2}$
- Seq: 28, 64, 144, 320, 704, 1536, 3328, 7168, 15360, 32768, . . .
- OEIS A045623: Number of 1's in all compositions of $k+1$ for $n \geq 2$.



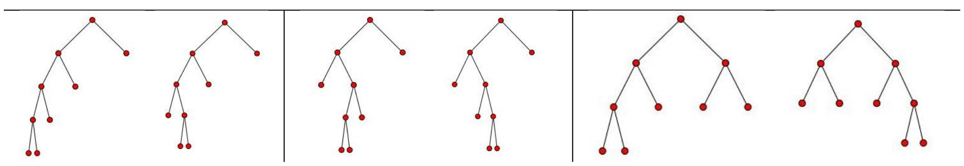
Class M

- $gf_{t_1,t_2}(\cdot) = \frac{x-2x^2+x^3}{1-3x+2x^2-x^3}$
- Seq: 28, 65, 151, 351, 816, 1897, 4410, 10252, 23833, 55405, . . .
- OEIS A034943: Binomial transform of Padovan sequence A000931for $n \geq 1$.



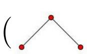
Class N

- $gf_{t_1,t_2}(\cdot) = \frac{x-x^2-x^3}{1-2x-x^2}$
- Seq: 29, 70, 169, 408, 985, 2378, 5741, 13860, 33461, 80782, ...
- OEIS A000129: Pell numbers: $a(0) = 0, a(1) = 1$; for $k \geq 1$, $a(k) = 2 \cdot a(k-1) + a(k-2)$ for $n \geq 2$.

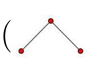


6.2 Conjecture for n Tree Avoidance

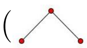
Let $gf_{t_1,t_2,t_3,\ldots t_{k-1},t_k}$ be the generating function whose coeffieient of the x^n term is the number of n-leaf trees avoiding the full binary trees $t_1, t_2, t_3, \ldots t_{k-1}, t_k$. Then

$$gf_{t_1,t_2,t_3,\ldots t_{k-1},t_k}(\cdot) = x + gf_{t_1,t_2,t_3,\ldots t_{k-1},t_k}($$  $$).$$

Now consider

$$gf_{t_1,t_2,t_3,\ldots t_{k-1},t_k}($$  $$).$$

Let S be the set of properties of avoiding each of the entire tree patterns $t_1, t_2, t_3, \ldots t_{k-1}, t_k$ while P is the set properties of avoiding not only the k entire tree patterns, but also to avoid all of the tree patterns on either the left or right of the tree. For each possible $J \subseteq P$ where $S \subseteq J$ and $gf_A \cdot gf_B$ is the way to satisfy the properties of J and avoid the chosen left subtrees on the left of the tree (in gf_A) or the chosen right subtrees on the right of the tree (gf_B). We conjecture that

$$gf_{t_1,t_2,t_3,\ldots t_{k-1},t_k}($$  $$) = \sum_{S \subseteq J \subseteq P} (-1)^{|J|-|S|} \cdot gf_A \cdot gf_B.$$

7 Conclusion

Throughout this paper, we have investigated non-consecutive pattern avoidance in binary trees. First we computed the avoidance generating functions for tree patterns with ≤ 5 leaves through exhaustion. Then we generalized this process by developing a recursive algorithm. Through analysis of this algorithm we proved that our avoidance generating functions are always rational. These avoidance generating functions were then related to recurrences and numerous patterns emerged. We then focused our research on avoiding two trees simultaneously.

Areas for further research include:

1. Finding more relationships between pattern-avoiding binary trees and other combinatorial objects.
2. Proving our conjecture for the avoidance generating function algorithm for n trees.
3. Investigating the structure behind the recurrences for $n \geq 6$,.
4. Finding the tree classes for avoiding two trees where one tree has more than 5 leaves.

References

[1] Eric S. Rowland, Pattern Avoidance in Binary Trees, *Journal of Combinatorial Theory*, Series A **117** (2010) 741-758.

[2] The On-Line Encyclopedia of Integer Sequences, published electronically at http://oeis.org, 2011.