“M” Arc Diagrams and Their Relationship with Borel’s Triangle

Kameron McCombs

May 9, 2017

Abstract

The Catalan Numbers famously count many combinatorial sets including arc diagrams on $2n$ points and binary trees on n vertices. In this paper, I show that there is a bijection between extended leaf marked binary trees on n unmarked vertices and k marked vertices and "m" arc diagrams on $2n-k$ points by creating two functions that map between extended leaf marked binary trees and "m" arc diagrams. I then show that these functions are inverse bijections.

1 Introduction

The Catalan Numbers are known for counting a large number of sets of objects. For example, they count both binary trees on n vertices and ways to arrange non-intersecting arcs over $2n$ points in the plane. Of course if the same number counts both of these sets of objects, then there must be the same amount of them. Then we should be able to find bijections between them.

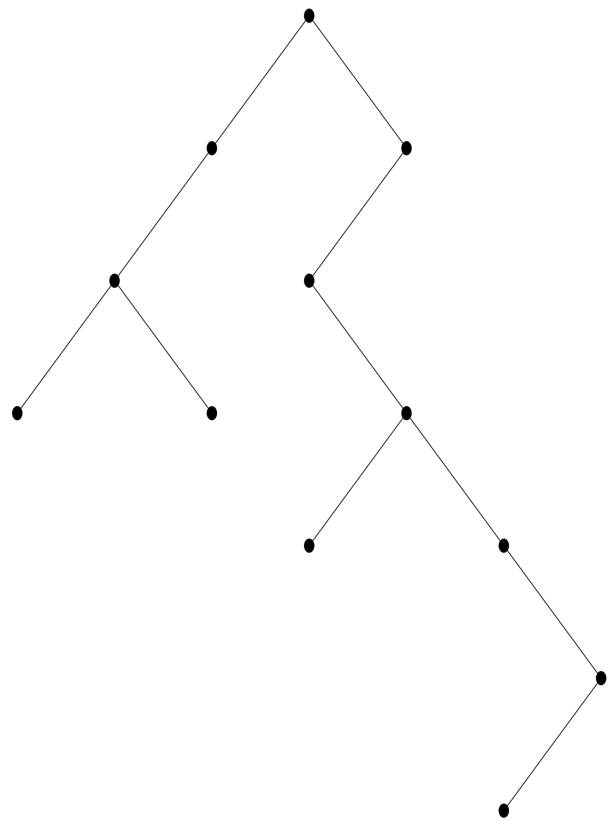
Definition 1.1. The *nth* Catalan Number is $\frac{1}{n+1}\binom{2n}{n}$.

There is also what is known as Catalan's triangle, which is a tool for computing the Catalan Numbers. Below is an example of Catalan's triangle. The Catalan Numbers appear on the diagonal of the diagram.

1			
1	1		
1	2	2	
1	3	5	5

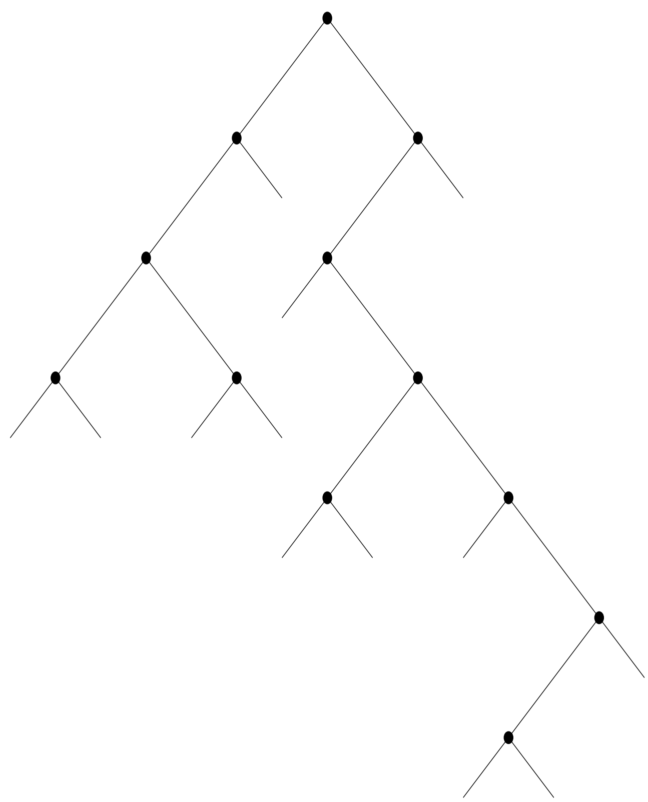
We will now consider a few of the sets that are counted by the Catalan Numbers.

Definition 1.2. A binary tree on n vertices is an acyclic connected graph with n nodes, each with 0, 1, or 2 children connected by descending to the left or to the right, including one node known as the root vertex which is not the child of any other node. Each child is identified as either the left or the right child of a parent node, where a node is a right child if it descends down and to the right and a left child if it descends down and to the left of the parent node. Finally, a leaf node is a vertex without children.



This is a running example of a binary tree on 12 vertices.

Definition 1.3. A balanced binary tree on $2n + 1$ vertices is a binary tree where each vertex has either 0 children or 2 children.



The corresponding balanced binary tree on 25 vertices.

Theorem 1.1. *The trees on n vertices and balanced binary trees on $2n+1$ vertices are in bijection.*

Proof. The bijection is defined as follows: Given a tree, add children to each vertex until it has two children. Given a balanced binary tree, delete all of the leaves. These are inverse operations, and so there must be a bijection between the two sets. $\square$

Definition 1.4. A set of parentheses is balanced if each left parenthesis has a corresponding right parenthesis, and from left to right there are never more right parentheses than left parentheses.

(((())()))(()(())()(()))
Here is an example of a set of 12 pairs of balanced parentheses.

Definition 1.5. A triangulation of a convex polygon is a division of said polygon into triangles by adding edges between vertices.

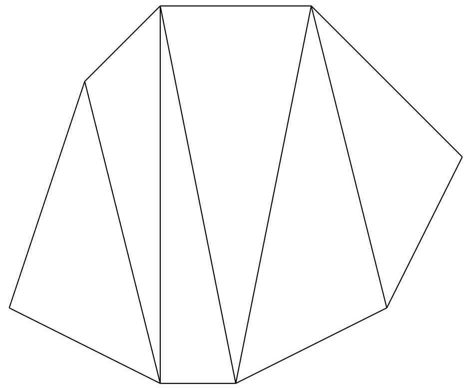


Figure 1: A triangulation of a convex octagon.

Definition 1.6. An arc diagram is a set of $2n$ points on a horizontal line, where each pair of points is connected via n non-intersecting arcs.

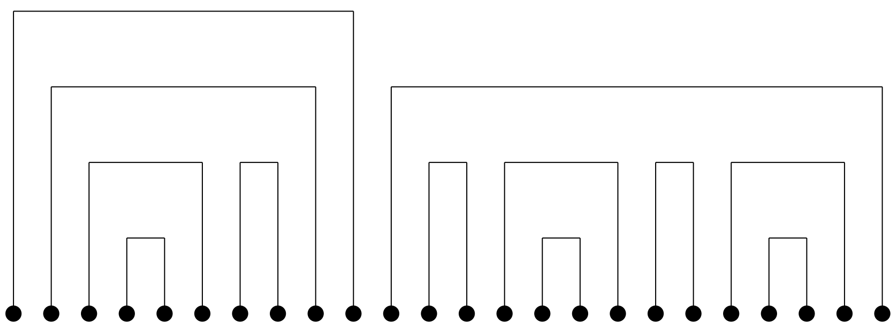


Figure 2: An arc diagram on 24 points.

Theorem 1.2. *C_n is equal to the number of arc diagrams on $2n$ points.*

Theorem 1.3. *There are 214 sets known to be counted by the Catalan Numbers [2].*

Theorem 1.4. *Arc diagrams on $2n$ points are in bijection with binary trees on n vertices.*

Proof. We define inverse bijections as follows. Let f_b be a function that maps from binary trees to arc diagrams. Define f_b through this process: First take any binary tree T on n vertices. Then replace it with its corresponding balanced tree $T*$. Then arrange a horizontal line of $2n$ vertices. Now perform a standard depth first traversal, which is defined as follows. Starting at the root vertex move to the left child whenever possible one vertex at a time. If it is impossible to move to a left child, go back up the tree until it is possible to move to a right child and do so. With that being said, construct the arc diagram as follows:
1. If the vertex you are at is a left child, open an arc on the corresponding arc diagram.
2. If the vertex you are at is a right child, close the nearest opened arc.
Now define a function f_a from arc diagrams on $2n$ points to binary trees on n vertices by this process. Given any arc diagram M, start a tree by making a root vertex. Then start at the leftmost vertex of the arc diagram and move right. At each point follow these rules to construct the tree:
1. If the point opens an arc, then create a left child from the current vertex.
2. If the point closes an arc, then create a right child from the parent of the corresponding point that opened the arc.

After traversing every point in the arc diagram, take the tree and delete all of the leaves. What is left is a binary tree on n vertices.
Observe that these two functions are inverse bijections. $\square$

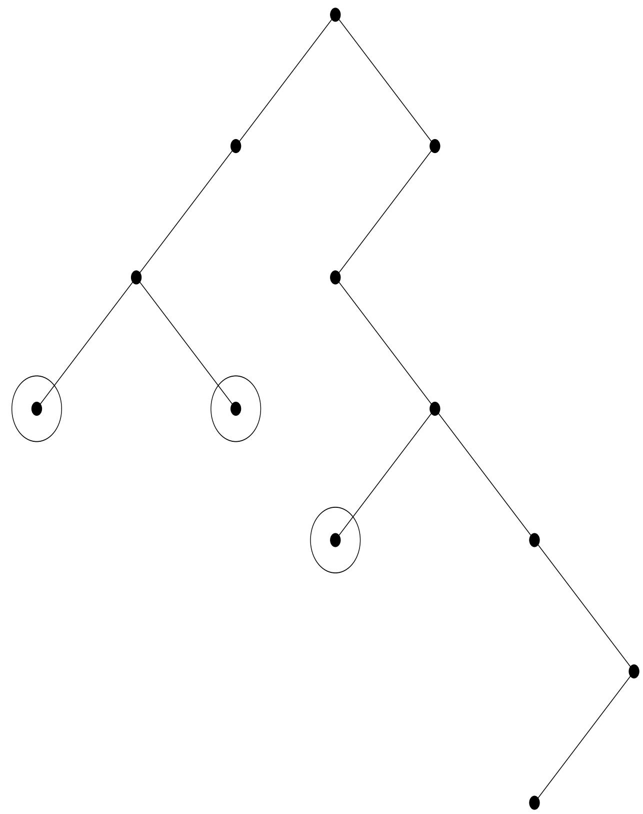
2 Leaf Marked Trees and Borel's Triangle

In a recent paper [1] Francisco, Mermin, and Schweig tweaked the objects mentioned above and bijections still held. We will show that tweaking arc diagrams into a new object yields the same count.
Borel's triangle is a tool similar to Catalan's triangle that is used to count these new sets. Here is an example of Borel's triangle.

1			
2	1		
5	6	2	
14	28	20	5

Definition 2.1. The rightmost leaf of a binary tree is the leaf obtained by descending and taking a right path whenever possible.

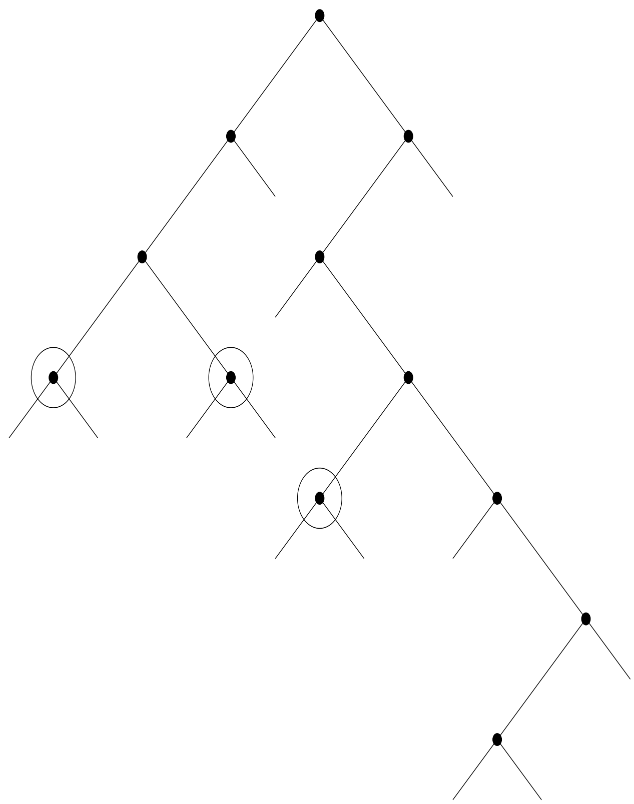
Definition 2.2. A leaf marked tree is a pair (T, X) where T is a tree, and X is a (possibly empty) subset of its leaves such that the rightmost leaf is not in X.



The previous binary tree with 9 unmarked vertices and 3 marked vertices.

Definition 2.3. An extended leaf marked tree T' is the corresponding balanced tree to a leaf marked tree T.

Observation: Again adding the leaves yields no new information or changes to the existing structure. It is clear that there is a bijection between leaf marked trees with n unmarked vertices and k marked vertices and extended leaf marked trees with $2n + k + 1$ unmarked vertices and k marked vertices.



The extended leaf marked tree corresponding to the previous leaf marked tree.

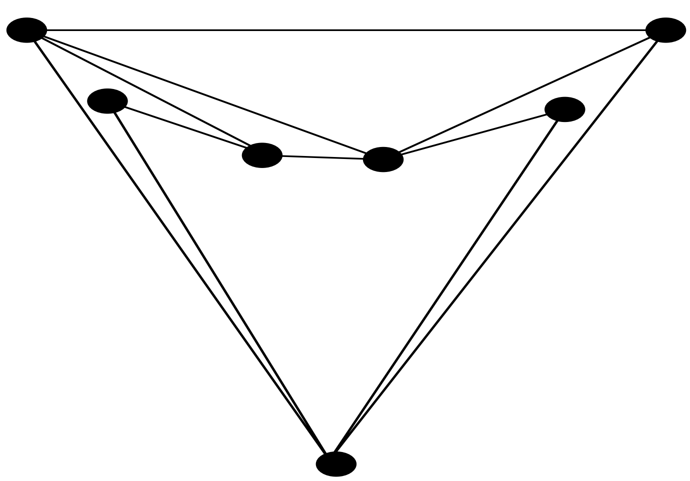
Definition 2.4. A pseudo-triangle is a plane polygon with exactly three convex angles.

Definition 2.5. A pseudo-triangulation of a point configuration is a subdivision of the point configuration into pseudotriangles.

Definition 2.6. A vertex in a pseudo-triangulation is pointed if it has a concave angle.

Definition 2.7. A pseudo-triangulation is pointed if every vertex is pointed.

Definition 2.8. The single chain of length n is $n+1$ points on a circular arc, together with a single point outside of the arc that can see all of the other points.



A pointed pseudo-triangulation.

Definition 2.9. An m arc diagram on $2n+k$ vertices is a set of $2n-k$ horizontal points lying in the plane such that there are $n-2k$ non-intersecting arcs connecting 2 points and k non-intersecting arcs connecting 3 points in an "m" that also satisfy these conditions:
1. There cannot be an arc underneath the right side of an "m" figure.
2. There must be one full arc between an "m" figure and the end of the diagram.

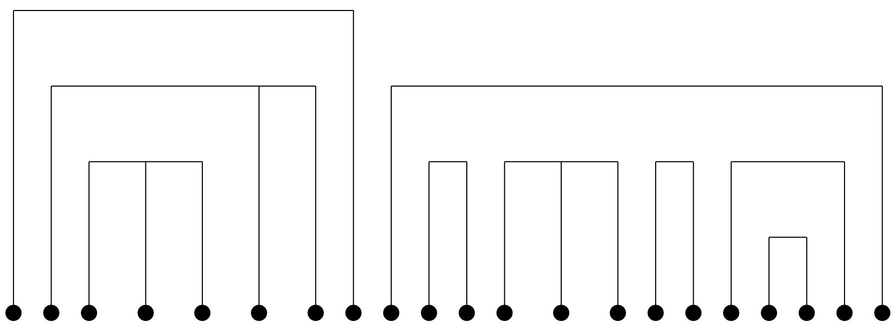


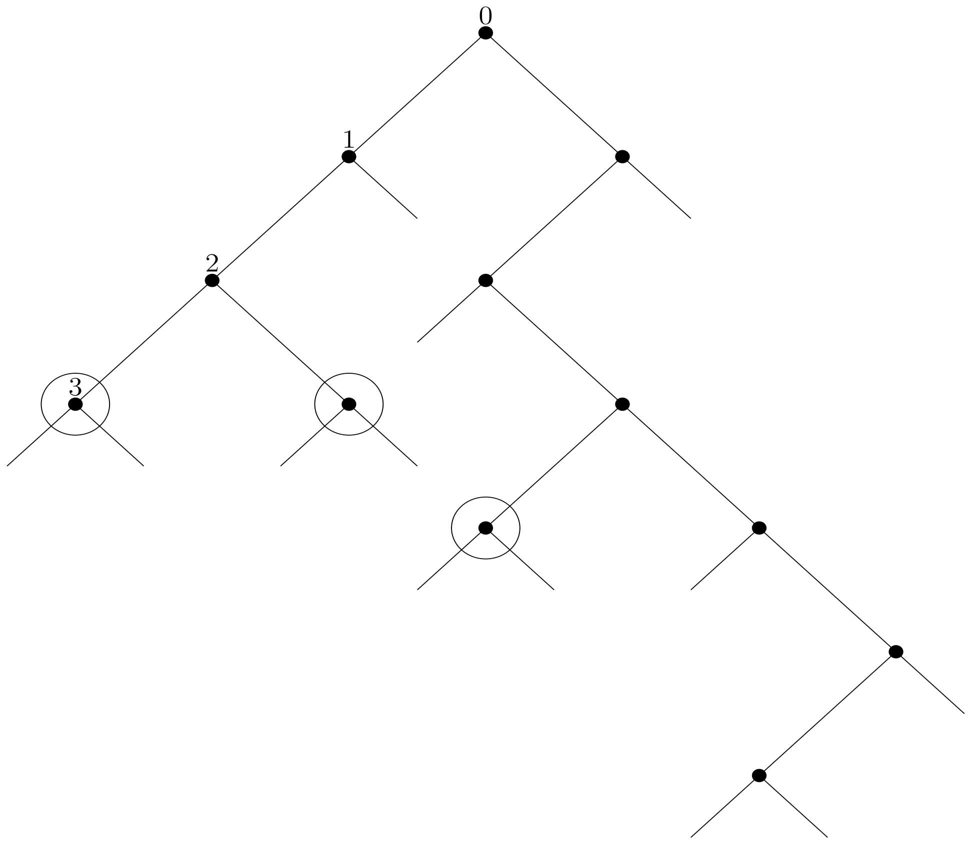
Figure 3: An "m" arc diagram on 18 regular legs and 3 "m" legs.

Our main result is the following:

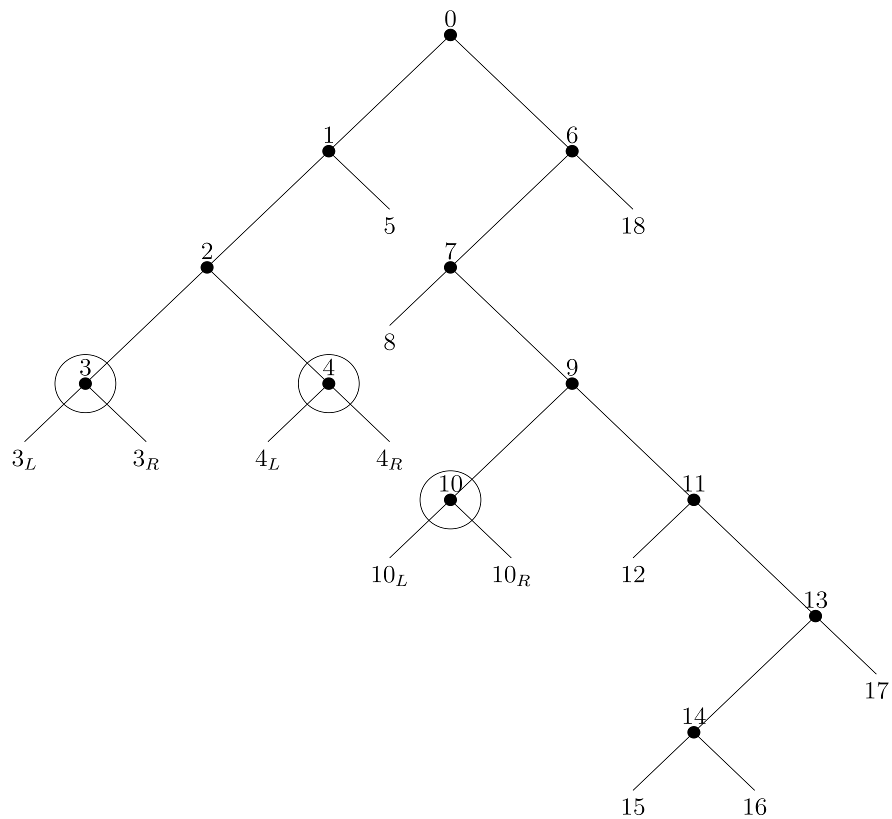
Theorem 2.1. *There is a bijection between leaf marked binary trees on $n+k$ vertices where n vertices are unmarked and k leaves are marked and m arc diagrams on $2n+k$ points. In particular, "m" arc diagrams are in bijection with the other sets counted by Borel's triangle.*

We begin the proof of Theorem 3 by defining a vertex labeling on both marked trees and "m" arc diagrams.

Starting with an extended leaf marked tree T on $n+k$ vertices(n umarked and k marked), label the root vertex 0. Then walk the tree in a standard depth-first traversal, doing the following:
1. Label each vertex with an integer in the order that they are encountered, skipping the children of marked vertices.
2. For each marked vertex (labeled s), label its children s_L and s_R.



The beginning of the labeling of the extended leaf marked binary tree.



The complete labeling of the extended leaf marked binary tree.

Observation: There is a bijection between extended leaf marked binary trees and labeled extended leaf marked binary trees. Apply the labeling to any extended leaf marked binary tree. Then delete the labeling from the tree. It is clear that these are inverse operations.

Lemma 2.1. *If the vertex labeled s is a leaf, then the vertex labeled $s+1$ is a right child.*

Proof. Suppose that the vertex labeled $s+1$ is a left child. Then as defined in the labeling steps, the vertex labeled s must be the parent of the the vertex labeled $s+1$. This is a contradiction because we assumed the vertex labeled s was a leaf. Therefore, the vertex labeled $s+1$ is a right child.

□

Lemma 2.2. *If the vertex labeled s is marked, then the vertices labeled s_L, s_R are leaves. In particular, s_L is a left child leaf and s_R is a right child leaf.*

Now define a labeling on an "m" arc diagram with $2n+k$ points on a horizontal line as follows:
1. Starting from left to right label the points 1 to $2n$, skipping over the middle legs of any "m" figure.

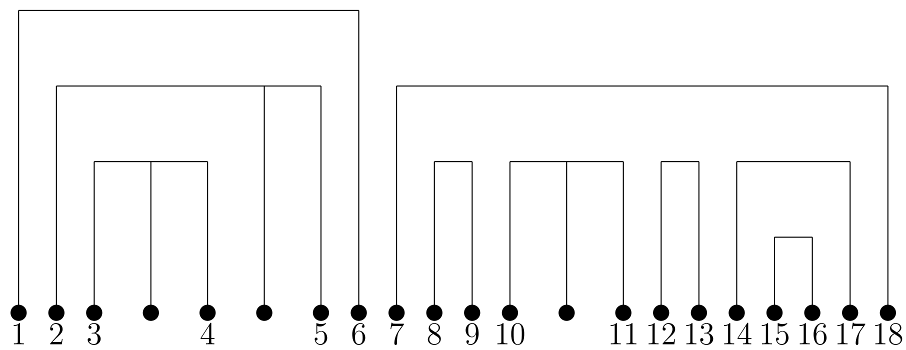


Figure 4: The labeling of the previous “m” arc diagram on 9 regular legs and 3 “m” legs.

2. If the point is the middle if an “m” and the point directly to the left is labeled s, then label the point s'.

Remark: There cannot be an instance where there are two primes in a row. This would imply that there is an ”m” arc with four legs, which would mean we are labeling something that is not an ”m” arc diagram.

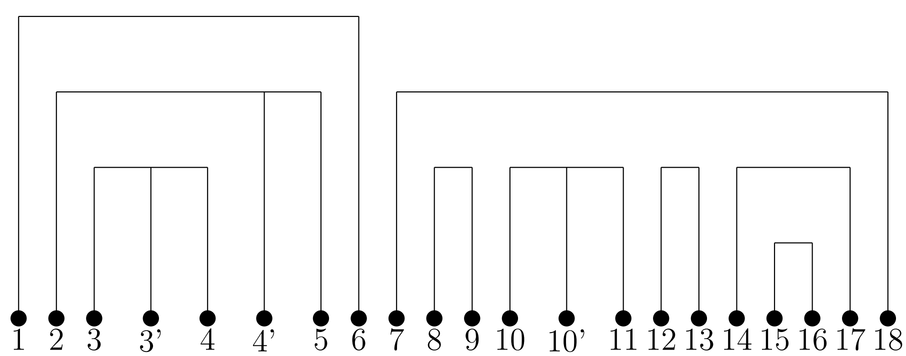


Figure 5: The complete labeling of the previous “m” arc diagram on 18 regular legs and 3 “m” legs.

Lemma 2.3. *There is a bijection between labeled “m” arc diagrams and unlabeled “m” arc diagrams.*

Proof. Apply the labeling on an “m” arc diagram. By just deleting the labeling, it is clear that this is an inverse to the labeling scheme. □

Define a function f_T from labeled extended leaf marked binary trees on n unmarked vertices and k marked vertices to labeled “m” arc diagrams on $2n + k$ vertices as follows:

1. Create a set of points labeled 1 through n on a horizontal line. Start at the point labeled 1.
2. At the point labeled s, if the vertex on the tree labeled s is a left child, then open an arc from that point.
3. If the vertex labeled s is a right child, then connect the point labeled s to the arc opened from the point labeled t, where t is the labeling of the corresponding left child to the right child labeled s.
4. If the vertex labeled s on the tree is a marked child, then create a new point on the “m” arc diagram labeled s' immediately to the right of the point labeled s. From this point create a sub-arc connecting s' to the nearest arc above it.

Remark: In step 3 it is guaranteed that there will be a corresponding left child because the tree is balanced.

Define a function f_M from labeled "m" arc diagrams on $2n+k$ points to labeled extended leaf marked binary trees on n unmarked vertices and k marked vertices as follows:

1. Start constructing an extended leaf marked binary tree by creating a root vertex and labeling it 0.
2. Going from left to right on the "m" arc diagram, if the point labeled s opens an arc, create a left child to the vertex labeled $s-1$ on the tree. Label this vertex s.
3. If the point is labeled s', then mark the vertex labeled s on the tree and give it a left child labeled s_L and a right child labeled s_R.
4. If the point labeled s closes an arc or "m" with leftmost point labeled t, create a right child from the vertex labeled $t-1$ on the tree and label the right child s.

In the series of Lemmas below, we prove that f_M, f_T are inverse bijections.

Lemma 2.4. *Suppose $T*$ is a labeled extended marked binary tree. Let $s \in T*$ be unmarked with children $s+1, t$. Then in $f_T(T*)$ there is an arc from $s+1$ to t.*

Proof. Let $T*$ be a labeled extended marked binary tree and let $s \in T*$ be an unmarked vertex with children $s+1, t$. First note that $s+1$ must be a left child because of the labeling scheme. Then t must be a right child. Let $M* = f_T(T*)$. Then in $M*$, we have that t must close an arc that is connected to the point labeled $s+1$ by construction.

□

Lemma 2.5. *Let $M*$ be a labeled "m" arc diagram, and let there be an arc from s to t, with $s, t \in M$. Then in $f_M(M*)$ the vertices labeled s, t are the left and right children, respectively, of the vertex labeled $s-1$, which is unmarked.*

Proof. Let $M*$ be a labeled "m" arc diagram, and let there be an arc from s to t, with $s, t \in M$. Let $T* = f_M(M*)$. Then in $T*$, we have that the vertex t is the right child of the vertex $s-1$, and further the vertex s is the left child of $s-1$. This is immediate from the definition of f_M.

□

Lemma 2.6. *Lemma 3: Let $T*$ be a labeled extended marked binary tree. Let $s \in T$ be a marked left child. Then $F_T(T*)$ has a vertex labeled s' connected to s immediately to the right of s.*

Proof. Let $T*$ be a labeled extended marked binary tree. Let $s \in T$ be a marked left child. Let $M* = f_T(T*)$. If $s \in T*$ is marked, then it has two children s_L and s_R. In $M*$, the two children will be condensed into a single point s' that will be between s and $s+1$ that is connected to s.

□

Lemma 2.7. *Lemma 4: Given a tree T the image $f_T(T)$ is an m-arc diagram. In other words, none of the arcs in the image intersect.*

Proof. Suppose for a tree T and its image $f_T(T)$, that there are arcs from s_1 to s_2 and from t_1 to t_2 that intersect. Without loss of generality, assume that t_1 is between s_1 and s_2. Then t_1 being underneath the arc from s_1 to s_2 would imply that t_1 is a descendent of s_1. As t_2 is to the right of s_2, then in the preimage, t_2 is either a descendent of s_2 or the right child of another vertex above t_2. This is a contradiction, as t_2 must be the corresponding right child to the left child t_1. Then there cannot be intersecting arcs in the image of T. □

Lemma 2.8. *Lemma 5: Let $T*$ be a labeled extended marked binary tree. Let $s \in T*$ be a marked right child. Then $F_T(T*)$ has a vertex labeled s' immediately to the right of s that is connected to $s+1$.*

Proof. Let $T*$ be a labeled extended marked binary tree. Let $s \in T*$ be a marked right child. Let $M* = f_T(T*)$. If s is a marked vertex, then it has two children s_L and s_R. In $M*$, these two children will be condensed into a single point s' that will be placed between s and $s+1$. The point will also connect to $s+1$ by construction.

□

Lemma 2.9. *Lemma 6: Let $T*$ be a labeled extended marked binary tree. Let $s \in T*$ be a leaf that is not the rightmost leaf. Then in $f_T(T*)$, $s+1$ closes an arc.*

Proof. Let $T*$ be a labeled extended marked binary tree and let $M* = f_T(T*)$. Let $s \in T*$ be a leaf that is not the rightmost leaf. If s is a left leaf, then it must have a corresponding right leaf $s+1$. Then in $M*$, the vertex labeled $s+1$ closes an arc with s by definition of f_T. If s is a right leaf, then the vertex labeled $s+1$ must be a right vertex. Then that vertex must have a corresponding left vertex. Let the label on that vertex be t. Then in $M*$, the vertex $s+1$ closes an arc with the vertex t by definition of f_T. As any non-root vertex in $T*$ is either a left child or right child, this ends the proof.

□

Lemma 2.10. *Lemma 7: Let $M*$ be a labeled "m" arc diagram. Let $s+1 \in M*$ be a vertex that closes an arc with t. Then in $f_M(M*)$, the vertex s is a leaf.*

Proof. Let $M*$ be a labeled "m" arc diagram and let $T* = f_M(M*)$. Let $s+1 \in M*$ be a vertex that closes an arc with t. By the definition of the function, a right child will be added to vertex $t-1$ in $T*$, which will be labeled $s+1$. Suppose the vertex labeled s is not a leaf. Then it must have a right and left child because the tree is balanced. This would imply that there is a vertex in $M*$ directly to the right of s that opens an arc. This point must be $s+1$ because there is no other point directly to the right of s. This would imply that the vertex labeled $s+1$ in $T*$ is a left child. This is a direct contradiction to the vertex being a right child. Therefore, the vertex s in $T*$ must be a leaf.

□

Lemma 2.11. *Lemma 8: Let $M*$ be a labeled "m" arc diagram. Let $s' \in M$. Then in $f_M(M*)$, s is marked, and its children s_L, s_R are leaves.*

Proof. Let $M*$ be a labeled "m" arc diagram. Let $s' \in M$. Finally, let $T* = f_M(M*)$. In $T*$, the definition of the function f_M states that the existence of s' means that the vertex s is marked. Then a left child s_L and a right child s_R are added to that vertex. Suppose these two vertices are not leaves. This would mean $M*$ has a vertex after s' that opens an arc and a vertex after s' that closes an arc before closing the "m". This is a contradiction as $M*$ would not be a labeled "m" arc diagram. Then the two vertices must be leaves.

□

Theorem 2.2. *Theorem: f_T and f_M are inverse bijections.*

Proof. Consider a labeled extended leaf marked tree with n unmarked vertices and k marked vertices. We will attempt to show that the structure of any labeled extended marked tree $T*$ and any labeled "m" diagram is unchanged in the image of $f_M \circ f_T$ and $f_T \circ f_M$ respectively. In other words, f_T has left and right inverse f_M. First note that the root vertex will always remain unchanged. There are then 4 cases: when the vertex is a left child, when the vertex is a right child, and when the vertex is the left child of a marked vertex, and when the vertex is the left child of a marked vertex.
case 1: Suppose a vertex labeled s is a left child. Then the function f_T would map this vertex to the point labeled s on the corresponding "m" arc diagram. In the diagram that point would open

an arc. By the definition of the function f_M, we have that this point would map to the left child of the vertex labeled $s-1$. This is consistent with how the vertex labeled s was originally labeled, so it is unchanged.
case 2: Suppose a vertex labeled s is a right child. Then the function f_T would map this vertex to the point labeled s on the corresponding "m" arc diagram. In the diagram that point would close an arc with the point labeled t, where t is also the labeling of the corresponding left child. If we put the resulting "m" arc diagram through the function f_M we see that by construction the vertex labeled s is the right child of the vertex labeled $t-1$, where t is the label of the left child of $t-1$. Note that we are guaranteed that this happens because in the extended tree any right child has a corresponding left child. This is the exact same labeling that the vertex originally had so it is unchanged.
case 3: Suppose a vertex labeled s_L is the left child of a marked vertex labeled s. Then the f_T creates a point between points s and $s+1$ labeled s' on an "m" arc diagram. Going back through f_M we see that the point labeled s' is split into two vertices s_L and s_R which are the left and right children of marked vertex s. Then s_L is the labeling on the left child of marked vertex s, which means the labeling is unchanged.
case 4: Suppose a vertex labeled s_R is the right child of a marked vertex labeled s. Then f_T creates a point between the points s and $s+1$ labeled s' on the "m" arc diagram. The function f_M then takes the point s' and creates two vertices s_L and s_R, which are the left and right children of the vertex labeled s on the tree. This is exactly what was labeled s_R to begin with, so the labeling is unchanged.
Then we have that f_T is a right inverse of f_M.
Now suppose we have an "m" arc diagram on $2n+k$ points, with k points marked as primes. There are three cases: the point opens an arc, the point closes an arc, or the point forms the middle of an "m".
Case 1: Suppose the point labeled s opens an arc. Then under the function f_M it will map to the left child of a vertex labeled $s-1$. The function f_T will map this vertex back to the point to the right of the point labeled $s-1$(or to the first point if $s=1$). This point will also open an arc, so the labeling will remain unchanged.
Case 2: Suppose the point labeled s closes an arc with the point labeled t. Then the function f_M will map this point to the right child of the vertex labeled $t-1$. The function f_T will map this vertex to the point labeled s and that point will close an arc with the point labeled t, which means the labeling is unchanged.
Case 3: Suppose the point is labeled s'. Then it forms an "m" with the point s or $s+1$. The function f_M will send this point to the children labeled s_R and s_L on the marked vertex s. The function f_T will map these children to the point on the "m" arc diagram labeled s' that forms an "m" with the point labeled s. Then the labeling is unchanged.
Therefore, f_T is a left inverse of f_M, and the functions f_T and f_M are inverse bijections.

□

References

[1] Christopher A. Francisco, Jeffrey Mermin, and Jay Schweig. Catalan numbers, binary trees, and pointed pseudotriangulations. *European J. Combin.*, 45:85–96, 2015.

[2] Richard P. Stanley. *Catalan numbers*. Cambridge University Press, New York, 2015.