Skew Diagrams and Ordered Trees

Robert G. Rieper and Melkamu Zeleke

William Paterson University, Wayne, New Jersey

Received September 3, 2000; accepted December 10, 2000

DEDICATED TO DOMINIQUE FOATA

We use a known combinatorial argument to prove that among all ordered trees the ratio of the total number of vertices to leaves is two. We introduce a new combinatorial bijection on the set of these trees that shows why this must be so. Ordered trees are then enumerated by number of leaves, total path length, and number of vertices to obtain $q$-analogs of Catalan numbers. The results on ordered trees are then readily transferred by the skew diagrams to help enumerate parallelogram polyominoes by their area, perimeter, and other statistics.

1. INTRODUCTION

Ordered trees (often referred to as rooted plane trees or simply plane trees) are trees with a distinguished vertex called the root where the children of each internal vertex are linearly ordered. Ordered trees are drawn so that the children of each internal vertex are shown in order from left to right. The Catalan numbers $C_n = \frac{1}{n+1} \binom{2n}{n}$, $n \in N$, among other things, count the quantity of ordered trees with $n$ vertices [5, 11].

In this paper, we re-introduce some tree enumeration problems and introduce the parameter that provides the $q$-analogy of Catalan numbers using ordered trees. We avoid the use of Dyck paths and their grammars to obtain the recurrence relations of the generating functions we define and solve these recursions using a method that does not appear in the literature on tree enumeration. The method may be of its own interest in solving similar recursions.

1 To whom correspondence should be addressed: Department of Mathematics, William Paterson University, 300 Pompton Road, Wayne NJ 07470.
We begin with an elementary combinatorics problem whose solution provides the means to tackle more challenging problems in the subsequent sections. The elementary problem is the enumeration of ordered trees where we are only interested in the quantity of their vertices and how many of these might be leaves. This problem is well understood and its solution may be found in numerous places, but we are motivated to solve it again as the recurrence relation of the generating function used to solve the problem is similar to the one we used to obtain a $q$-analogy of Catalan numbers in Section 3.

Shapiro [9, 10] noted that the total number of leaves among these trees is half the total number of vertices and asked for a combinatorial proof of his observation. That is, there are five different ordered trees on four vertices and among the 20 vertices ten are leaves. These five trees are shown in Fig. 1 where the leaves have been distinguished by coloring them black and the others white. Note that the root is never counted as a leaf even though its degree may be one.

An immediate proof can be provided using generating functions. Let $C(x, y)$ be a generating function which enumerates ordered trees by number of leaves and vertices. Let $C_l, v$ be the number of ordered trees with $l$ leaves and $v$ vertices. Then

$$C(x, y) = \sum_{l, v} C_l, v x^l y^v$$

satisfies the following recursion (see Fig. 2).

$$C = y(C + xy) + y(C + xy)^2 + y(C + xy)^3 + \cdots$$

$$\Leftrightarrow C = y(C + xy) \left( \frac{1}{1 - (C + xy)} \right)$$

$$\Leftrightarrow C(1 - (C + xy)) = y(C + xy)$$

$$\Leftrightarrow C^2 + (xy + y - 1)C + xy^2 = 0.$$ 

Therefore,

$$C = \frac{1 - y - xy \pm \sqrt{(y + xy - 1)^2 - 4xy^2}}{2},$$

FIG. 1. The five ordered trees on four vertices.
FIG. 2. Generating ordered trees from others recursively. Choose a nontrivial tree (counted by $C$) or a leaf (counted by $xy$).

where we are obligated to choose the minus sign to produce a meaningful value of $C(0, 0)$. Hence,

$$C = \frac{1 - y - xy - \sqrt{(y + xy - 1)^2 - 4xy^2}}{2}.$$ 

We will return to the recursion (1) and its solution many times in subsequent sections where more challenging problems occur. For now, given an ordered tree with $v$ vertices we want to find

$$\frac{\text{total number of vertices}}{\text{total number of leaves}} = \frac{v[y^v]C(1, y)}{\sum [x^l y^v]C(x, y)}.$$ 

Observe that the right hand side is equal to

$$\frac{[y^v]yD_y C(1, y)}{[y^v](D_x C(x, y))|x = 1}.$$ 

Taking the derivative of $C(1, y)$ with respect to $y$ and the derivative of $C(x, y)$ with respect to $x$ we obtain

$$yD_y C(1, y) = -y + \frac{y}{\sqrt{1 - 4y}}$$ 

and

$$D_x C(x, y) = \frac{1}{2} \left( -y + \frac{y^2 + y - xy^2}{\sqrt{(y + xy - 1)^2 - 4xy^2}} \right).$$ (2)

Therefore,

$$\frac{[y^v]yD_y C(1, y)}{[y^v](D_x C(x, y))|x = 1} = 2.$$ 

The coefficient of $y^v$ in Eq. (2) gives the total number of vertices to be $\binom{2v - 1}{v - 1}$ (a known result) and we have proved the following result.
**Theorem 1.** Half the vertices among all ordered trees with $v$ vertices are leaves and equal in number to \( \frac{1}{2} \left( \frac{2^v - 1}{v - 1} \right) \).

Since the quantity of leaves and nonleaves must be equal in number, it is natural to ask if there exists a mapping of the set of ordered trees onto itself that maps a tree with $v$ vertices, $l$ of which are leaves, to one with $v - l$ leaves. If the mapping is bijective, then Theorem 1 is an immediate consequence. We are sure that this style of combinatorial proof is what Shapiro intended with his inquiry. Such a mapping already exists and is provided by a clever use of the bijection between ordered trees and binary trees given in Stanton and White [12]. In this mapping, leaves of the ordered tree become left leaves of the binary tree and nonleaves become right leaves. A mirror reflection of the binary tree and an application of the inverse mapping yields another ordered tree with the required property. Their bijection is not as natural nor as direct as we might wish so we provide another via skew diagrams in the next section.

2. **SKEW DIAGRAMS**

We define a mapping from ordered trees to another on $v$ vertices that encodes the trees.

Consider any ordered tree. Since the tree is imbedded rigidly in the plane, the notion of leftmost leaf is unambiguous. The leaf together with the root determine a unique path. In fact, each leaf of the tree can be described by its position, left to right, among all the leaves and there is a unique path from the root for each of them. If the path of the leftmost leaf has length $e$ (number of edges), then place $e$ dots along the first column (left to right) of the diagram. For each successive leaf in turn, we place dots equal in number to the path length along the next column. The rectangular array of dots obtained using this procedure from an ordered tree is called a skew diagram. Since two paths necessarily share the root vertex, we are assured of some overlap. The amount of overlap is equal to the number of vertices that the two paths have in common. Figure 3 illustrates the skew diagram of a small tree. To read the diagram and produce the conjugate tree we do the following. The first row (top to bottom) records information about the path of the leftmost leaf of the conjugate tree. In our example, there are two dots in the first row. Draw the root vertex and a path of length two proceeding from it downward and to the left. The next row has three dots so the next leaf terminates a path of three edges. This row overlaps the previous one in just one dot and hence the new path shares only the root vertex with the previous path. The last row has two dots and overlaps the previous one in two dots. The path to the new leaf then shares its first two
vertices with the previous path and proceeds from there downward and to the right.

In the example, the original tree has four leaves while its conjugate has three. Moreover, the skew diagram of the conjugate tree reproduces the original tree so that the mapping is bijective. Given an ordered tree on $v$ vertices, $l$ of which are leaves, we now convince ourselves that its conjugate has $l^* = v - l$ leaves. Draw a bounding rectangle about the skew diagram whose lower-left and upper-right corners are coincident with the first dot drawn and the last. The rectangle is $l$ dots wide and $l^*$ dots high with $l + l^* - 1$ diagonals. Since the number of diagonals equals the number of edges $v - 1$, we must have $l^* = v - l$.

The bijective mapping $T \leftrightarrow T^*$ given by the skew diagram thus produces an alternative proof of Theorem 1. However, no explicit mapping of the vertex set has been defined yet. Let $V$ and $V^*$ be the vertex sets of the tree and its conjugate, respectively. We now show that the skew diagram also produces a bijection $V \leftrightarrow V^*$. The bijection is accomplished via in-order traversal of $T$ and in-order, but decreasing traversal of $T^*$.

Choose a tree $T$ on $v$ vertices containing $e$ edges and draw its skew diagram. Draw diagonals proceeding from upper left to lower right on the skew diagram. Label the diagonals 1 through $e$ as shown in Fig. 4. Recall that the quantity of diagonals equals the number of edges $e$.

---

**FIG. 3.** A tree $T$ and its conjugate $T^*$.

**FIG. 4.** A labeling of $T$ and its conjugate $T^*$. 
To determine the label of a leaf in $T$ find the label $d$ of the diagonal containing the topmost dot in the column that corresponds to the leaf in the skew diagram. Since the number of vertices in any tree is one more than the number of edges, we see that the leaf receives label $d + 1$. For example, to find the label of the second leaf in $T$ look at the topmost dot in the second column. The label number of the diagonal containing this dot is 3 and hence the label of the second leaf in $T$ is $3 + 1 = 4$.

Similarly, to determine the label of a leaf in $T^*$ find the label $d^*$ of the diagonal containing the leftmost dot in the row that corresponds to the leaf in the skew diagram of $T$. Since the labeling of $T^*$ is in decreasing order and the root receives the label $v$ all the time, we see that the label of a leaf in $T^*$ is $v - \{\text{the number of dots above the row}\}$, which is clearly equal to the label $d^*$ of the diagonal containing the leftmost dot in the row. For example, to find the label of the second leaf in $T^*$ look at the leftmost dot in the second row. The label number of the diagonal containing this dot is 2 and hence the label of the second leaf in $T^*$ is 2.

We combine these observations into the following two lemmas for a later reference.

**Lemma 2.** The label received by a leaf in $T$ is $d + 1$, where $d$ is the label of the diagonal containing the topmost dot in the column that corresponds to the leaf in the skew diagram.

**Lemma 3.** The label received by a leaf in $T^*$ is $d^*$, where $d^*$ is the label of the diagonal containing the leftmost dot in the row that corresponds to the leaf in the skew diagram of $T$.

**Theorem 4.** Let $T$ be an ordered tree on $v$ vertices and $T^*$ its conjugate via the skew diagram. Label the vertices of $T$ (including the root) with the integers $1, 2, \ldots, v$ using in-order traversal and label the vertices of $T^*$ using in-order traversal, but in decreasing order. Then vertex $i$ in $T$ is a leaf if and only if vertex $i$ in $T^*$ is a nonleaf.

**Proof.** Since the skew diagram provides a mapping from $T$ to $T^*$ and vice versa, it suffices to show that if vertex $i$ is a leaf in $T$ then the vertex labeled $i$ in $T^*$ is a nonleaf. Suppose not, then some leaf in $T$ and a leaf in $T^*$ must both receive the same label. Then by Lemmas 1 and 2 we see that $d + 1 = d^*$ which means that the topmost dot in some column occupies a diagonal that immediately precedes the diagonal of a leftmost dot in some row. Hence this leftmost dot cannot be in the same column of dots containing the topmost dot. It has to then be in a column next to the one that contains the topmost dot. Moreover, by the construction of skew diagrams, the row containing the leftmost dot cannot be below the row containing the topmost dot. Hence the leftmost dot is either above or in the same row containing the topmost dot. If the leftmost dot is in the row
above the one containing the topmost dot, then \( d^* \geq d + 2 \), contradicting our assumption. If it is in the same row, then we get a contradiction to the fact that it is the leftmost dot. In any of the possible cases we arrive at a contradiction, and hence if vertex \( i \) is a leaf in \( T \) then the vertex labeled \( i \) in \( T^* \) must be a nonleaf.

3. PATH LENGTH AND \( q \)-CATALAN NUMBERS

Each leaf in an ordered tree has a unique path from the root to the leaf. The path length of the leaf is the number of edges on its path. The total path length of the tree is the sum of these path lengths. The generating function \( C(x, y) \) introduced in Section 1 enumerates ordered trees by number of leaves (counted by \( x \)) and number of vertices (counted by \( y \)). We insert a third indeterminate, \( q \), to record the total path length of the ordered tree. Thus, \( C(x, q, y) \) enumerates ordered trees by number of leaves, total path length, and number of vertices. It will be seen that these agree with the \( q \)-Catalan numbers studied by Pólya [8] and Gessel [6] (see also [2–4, 7, 13]).

To accomplish the enumeration we again employ the recursion depicted in Fig. 2 to find the generating function \( C(x, q, y) \). Each choice to be made is between an existing tree counted by \( C \) or a new leaf which is now counted by \( xqy \). Note that choosing a new leaf adds one to the total path length explaining the presence of \( q \) in \( xqy \). If a tree with \( l \) leaves is chosen instead, then the path length of each of its leaves is increased by one so that the total path length is increased by \( l \). The total path length can be adjusted via the substitution \( x \rightarrow xq \) in \( C(x, q, y) \). The new total path length is then properly recorded in \( C(xq, q, y) \). The recursion for the generating function \( C \) is then (the presence of the factor \( y \) in each term is to count the new root vertex)

\[
C(x, q, y) = y(C(xq, q, y) + xqy) + y(C(xq, q, y) + xqy)^2 + \cdots
\]

\[
\Leftrightarrow C(x, q, y) = y(C(xq, q, y) + xqy) \frac{1}{1 - (C(xq, q, y) + xqy)}
\]

\[
\Leftrightarrow C(x, q, y) = -y + \frac{y}{1 - C(xq, q, y) - xqy}.
\]

(3)

We now solve the functional recursion (3). The statement of the solution uses the common \( q \)-symbol defined as \( (q; q)_n = (1 - q)(1 - q^2) \cdots (1 - q^n) \) and, in general, \( (a; q)_n = (1 - a)(1 - aq)(1 - aq^2) \cdots (1 - aq^{n-1}) \) (often abbreviated as \( (q)_n \) and \( (a)_n \), respectively).

**Theorem 5.** The generating function \( C(x, q, y) \) is given by

\[
C(x, q, y) = 1 - xy - \frac{\sum_{n=0}^{\infty}(-1)^nq^n(yq; q)_n}{\sum_{n=0}^{\infty}(-1)^nq^n(yq; q)_n} \frac{1}{(yq; q)_n(q; q)_n}
\]

(4)
or alternatively

\[ C(x, q, y) = \frac{y \sum_{n=0}^{\infty} (-1)^{n+1} q^{\binom{n+1}{2}} y^{n+1} q^{n+1} x^{n+1} / ((yq; q)_n(q; q)_n)}{\sum_{n=0}^{\infty} (-1)^n q^{\binom{n}{2}} y^n q^n x^n / ((yq; q)_n(q; q)_n)}. \]  

(5)

**Proof.** The solution to the recursion (3) is accomplished by implicitly defining a new function \( F(x, q, y) \) by

\[ 1 - C(x, q, y) - xy = \frac{F(x, q, y)}{F(xq, q, y)}. \]

Then

\[ 1 - C(xq, q, y) - xqy = \frac{F(xq, q, y)}{F(xq^2, q, y)}. \]

Therefore,

\[ C(x, q, y) = -y + \frac{y}{1 - C(xq, q, y) - xqy} \]

\[ \Rightarrow 1 - C(x, q, y) - xy = 1 - xy + y - \frac{y}{1 - C(xq, q, y) - xqy} \]

\[ \Rightarrow \frac{F(x, q, y)}{F(xq, q, y)} = 1 - xy + y - \frac{y}{F(xq, q, y)/F(xq^2, q, y)} \]

\[ \Rightarrow F(x, q, y) = (1 - xy + y)F(xq, q, y) - yF(xq^2, q, y). \]

The last equation is a second order linear \( q \)-difference equation and we seek a series solution of the form

\[ F(x, q, y) = \sum_{n=0}^{\infty} a_n(q, y)x^n. \]  

(6)

Substitution into the \( q \)-difference equation and a comparison of the coefficients of \( x^n \) leads to the relationship

\[ a_n = (1 + y)q^n a_n - ya_{n-1}q^{n-1} - ya_nq^{2n} \]

with recursive solution

\[ a_n = \frac{-yq^{n-1}}{(1 - yq^n)(1 - q^n)} a_{n-1}. \]  

(7)

Repeated application of Eq. (7) yields the explicit solution

\[ a_n = \frac{(-1)^n q^{\binom{n}{2}} y^n}{\prod_{k=1}^{n} (1 - yq^k)(1 - q^k)} a_0 \]

\[ = \frac{(-1)^n q^{\binom{n}{2}} y^n}{(yq; q)_n(q; q)_n} a_0. \]
The value of \(a_0\) is superfluous and taken to be one. Substituting this result into equation (6) we have

\[
F(x, q, y) = \sum_{n=0}^{\infty} \frac{(-1)^n q^{\binom{n}{2}} y^n}{(yq; q)_n(q; q)_n}
\]

and we are done.

The alternate form of the solution (Eq. (5)) is obtained from the first via a common denominator and simple algebra. We have chosen to write the solution this way for comparison with the results stated in [4].

Maple implementation of Theorem 5 shows that the number of ordered trees with 12 vertices, 8 leaves, and total path length 15, for instance, is 540.

To obtain the \(q\)-Catalan numbers as defined by Pólya [8] and explored further by Gessel [6], rewrite the recursion (3) in the form

\[
C(x, q, y) = xqy^2 + xqyC(x, q, y) + yC(xq, q, y) + C(x, q, y)C(xq, q, y)
\]

and introduce the functions \(C_n = C_n(x, q)\) by \(C(x, q, y) = \sum_{n \geq 0} C_n(x, q)y^n\).

A comparison of the coefficients of \(y^n\) leads to

\[
C_n(x, q) = xqC_{n-1}(x, q) + C_{n-1}(xq, q) + \sum_{k=0}^{n} C_k(x, q)C_{n-k}(xq, q),
\]

where \(C_0(x, q) = C_1(x, q) = 0\) and \(C_2(x, q) = xq\). These are the \(q\)-Catalan numbers of Pólya and Gessel (see [2]).

We conclude this section with an application of Theorem 5 to enumeration of parallelogram polyominoes, contiguous unit squares with vertices at integer points in the plane that have two nonintersecting paths with only north and east steps as their border. It is easy to see that the skew diagrams used earlier in this article are actually parallelogram polyominoes as illustrated in Fig. 5.

**Corollary 6.** The generating function \(PP(q, y)\) which enumerates parallelogram polyominoes by area \((q)\) and semi-perimeter \((y)\) is given by

\[
PP(q, y) = 1 - y - \frac{\sum_{n=0}^{\infty} (-1)^n q^{\binom{n}{2}} y^n / ((yq; q)_n(q; q)_n)}{\sum_{n=0}^{\infty} (-1)^n q^{-\binom{n+1}{2}} y^n / ((yq; q)_n(q; q)_n)}
\]

\[(8)\]

**FIG. 5.** Identifying a parallelogram polyomino with an ordered tree.
or alternatively

\[
PP(q, y) = \frac{y \sum_{n=0}^{\infty} (-1)^{n+1} q^{n(\frac{n}{2})} y^{n+1} q^{n+1}/((yq; q)_n(q; q)_n)}{\sum_{n=0}^{\infty} (-1)^n q^{\frac{n(n+1)}{2}} y^n q^n/((yq; q)_n(q; q)_n)}. \tag{9}
\]

Proof. \(PP(q, y) = C(1, q, y). \]

4. CONCLUSION

We conclude with some remarks concerning our solution of the functional recursion (2), the function \(F\) given in Eq. (5), and the \(q\)-difference equation (6) that it satisfies. Recursions similar to ours appear throughout the recent literature concerning parallelogram polyominoes and they may be solvable using the techniques herein. In fact, the recursion found by Delest and Fedou [4] is essentially identical with ours (we have one additional parameter to count vertices). Their recursion was obtained by a creative analysis using algebraic grammars and a correspondence between parallelogram polyominoes and Dyck paths. Since Dyck paths and ordered trees also correspond it is easy to see that the approaches are related.

The technique we used to solve the recursion appears briefly in the classic book on partitions by Andrews (p. 104 in [1]) and concerns the solution of continued fractions of a type investigated by Ramanujan. It is very easy to convert the functional recursion (2) into a continued fraction by its repeated application. We mention this here because of our interest in an infinite product representation of \(F\). We have not found such a product but Rogers–Ramanujan identities and others suggest that one might exist. Many thanks to R.P. Stanley for indirectly pointing us in this direction (see Exercise 6.34a and its solution in [11]).

ACKNOWLEDGMENTS

The second author acknowledges summer research support from the Center for Research of the College of Science and Health, William Paterson University.

REFERENCES