Parking Distributions on Trees

Steve Butler*

Department of Mathematics, Iowa State University, Ames, IA 50011, USA

Ron Graham

Dept. of Computer Science and Engineering, UC San Diego, La Jolla, CA 92093, USA.

Catherine H. Yan

Department of Mathematics, Texas A&M University, Station, TX 77843, USA

Abstract

We consider a generalization of parking functions to parking distributions on trees and study the unordered version and a q-analogue. We give an efficient way to form generating functions to compute these values and establish the positivity and log-concavity of a related polynomial. We also connect the unordered parking distributions on caterpillars (trees whose removal of leaves results in a path) to restricted lattice walks.

Keywords: parking functions, rooted trees, parking distributions, restricted lattice walks

2010 MSC: 05A15, 05C05, 05C30

1. Introduction

The classic parking function is defined by a parking process on a one-way street. Suppose there are \( n \) parking spaces labeled 1, 2,\ldots, \( n \) along the street. A set of \( n \) cars enter the street one by one, each with a preferred space. As each car enters it moves directly to its preferred space and parks there if the space is available. Otherwise the car moves towards the exit and takes the first available space. If there is no space available, the car exits without parking. A parking function of length \( n \) is a preference sequence for the cars in which all cars are able to park (not necessarily in their preferred spaces.)

We can represent this as a process on a path graph (see Figure 1) where the vertices correspond to the available spaces and the cars move from left to right. The preferences (1, 2, 3, 4), (2, 1, 3, 4) or (1, 2, 4, 1) all correspond to parking functions, while (2, 2, 2, 4) will have one car leave un-parked so is not a parking function. It is well-known that any permutation of a parking function is also a parking function.

![Figure 1: A path with the root at 4](image)

The set of parking functions is a basic object lying in the center of enumerative combinatorics, with many generalizations and connections to other research areas, such as hashing and linear probing in computer science, graph searching algorithms, interpolation theory, diagonal harmonics, and sandpile models. Because of their rich theories and applications, parking functions and their variations have been studied extensively in the literature. For a comprehensive survey on the combinatorial theory of parking functions see Yan [20].

There is a particular generalization of parking functions that has been considered recently. Returning to the parking process described at the beginning of this paper, the essential features are as follows:

*Corresponding author

Email addresses: butler@iastate.edu (Steve Butler), graham@ucsd.edu (Ron Graham), cyan@math.tamu.edu (Catherine H. Yan)
- Each car has an initial preferred parking location.
- If a location is currently occupied then cars have a consistent rule to proceed and search for an opening.
- Each car will terminate its search after finitely many steps and exit.

For the path we start at a vertex and then if we cannot find a spot we continue moving along the path and eventually exit. If we make the last vertex the root of the path, then the parking rule can be stated as follows:

*Go to the preferred vertex, if it is not available continue moving towards the root and park at the first available spot; if no spot has been found and the root is reached, then exit.*

This rule works for any rooted tree \( T \). This suggests that one can consider parking functions on a general rooted tree \( T \), where each vertex of \( T \) represents a parking space, and edges are oriented toward the root, representing one-way streets. Again \( n \) cars enter one by one, each with a preferred parking space. A sequence of preferences such that all cars can park in the vertices of \( T \) under the above rule is called a \( T \)-parking function and was first studied by Bruner and Panholzer [1], who also considered parking functions on mappings. Via analytic combinatorial techniques, Bruner and Panholzer revealed a beautiful relation between the total number of parking functions on rooted trees and that on mappings, presented exact and asymptotic enumerative results, and described a phase change behavior when one considers \( m \) cars parking on \( n \) available spots.

A variation of classic parking functions is the non-decreasing parking functions, which are parking preference sequences \( (a_1, a_2, \ldots, a_n) \) such that \( a_1 \leq a_2 \leq \cdots \leq a_n \). It is well-known that in the classical case, \( (i.e., \ T \ is a path with \ n \ vertices) \), there are \( (n + 1)^{n-1} \) parking functions and \( C_n \) non-decreasing parking functions, where \( C_n = \frac{1}{n+1} \binom{2n}{n} \) is the \( n \)-th Catalan number. In this paper we consider the generalization of non-decreasing parking functions on rooted trees, which are called \( T \)-parking distributions and can be represented by functions \( f : V(T) \to \mathbb{N} \), where \( \sum_{v \in T} f(v) = |T| \) with \( f(v) \) indicating the number of cars that prefer vertex \( v \), and with \( f \) corresponding to a preference assignment where all cars can be parked. Parking distributions are well-defined since permuting the order of a \( T \)-parking function does not affect whether all cars are able to park. We will also consider a \( q \)-analogue of the number of parking distributions which keeps track of the total number of failed attempts to park, or the *bumps*. Again this number is independent of the order the cars attempt to park.

**Example 1.** The rooted tree \( T \) shown in Figure 2 has twelve \( T \)-parking distributions, which we can group by the number of failed parking attempts (“bumps”).

<table>
<thead>
<tr>
<th>bumps</th>
<th>parking functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>12345</td>
</tr>
<tr>
<td>1</td>
<td>11245, 12245, 12334, 12344</td>
</tr>
<tr>
<td>2</td>
<td>11234, 11244, 12234, 12244</td>
</tr>
<tr>
<td>3</td>
<td>11124, 11224, 12224</td>
</tr>
</tbody>
</table>

Therefore the \( q \)-analogue form of the \( T \)-parking distribution is \( 1 + 4q + 4q^3 + 3q^3 \) (i.e., the coefficient of \( q^i \) is the number of times we have exactly \( i \) bumps).

We proceed as follows. In Section 2 we characterize the set of \( T \)-parking distributions by a system of \( n \) linear inequalities. In Section 3 we introduce generating functions related to parking distributions which contain additional information to aid in the quick computation of these functions and establish some basic results including the positivity and log-concavity of a related polynomial. In Section 4 we consider parking distributions on caterpillars (trees where removing all leaves yields a path) and give a connection to restricted lattice walks. Finally in Section 5 we give some concluding remarks and open questions.

---

1 We started this project in the summer of 2013 before [1] appeared in the arXiv. We had considered the parking process on rooted trees independently, but not on mappings.
2. Characterization of parking distributions

The aim of this section is to give a characterization for parking distributions. We start with $T$-parking functions. Let $T$ be a rooted tree with vertices labeled 1, 2, ..., $n$. Let $c_1, c_2, ..., c_n$ be the $n$ cars and assume that $c_i$ prefers the vertex with label $u_i$. The sequence $(u_1, ..., u_n)$ is a $T$-parking function if and only if all the cars can find a parking space in $T$.

It is easy to check that if $(u_1, ..., u_n)$ is a $T$-parking function, then so is $(u_{\sigma_1}, ..., u_{\sigma_n})$ for any permutation $\sigma$ of length $n$. A proof can be found in Bruner and Panholzer [1, Lemma 2.1]. Below we prove a stronger result in Theorem 1.

The permutation invariance allows us to view the parking process as a special ball-bin distribution problem. We start with $n = |T|$ distinguishable balls each having a preferred vertex and on each vertex of $T$ there is a bin that can hold one ball. Now place all balls according to their preferences. In any bin that has more than one ball have any extra balls move toward the root. A $T$-parking function is then an assignment of preferences where no ball drops out of the root. The $T$-parking distributions correspond to the cases where the balls are indistinguishable. This ball-bin model will help us understand the behavior of parking distributions.

Recall that a characterization of classical parking functions of length $n$ is that there are at least $i$ terms less than or equal to $i$ in the preference sequence. For tree parking functions and parking distributions, we have a similar characterization.

In a rooted tree $T$, let $T_u$ be the subtree of $T$ rooted at $u$, i.e., $T_u$ contains all the descendants of $u$, which are vertices $w$ such that $u$ lies in the unique path from $w$ to the root. A distribution of indistinguishable balls on the vertices of $T$ can be described as a function $f: V(T) \rightarrow \mathbb{N}$, where $f(u)$ is the number of balls preferring the vertex $u$.

**Theorem 1.** Let $f$ be a distribution of indistinguishable balls on the vertices of $T$. Apply the parking process to the balls. Then each vertex of $T$ will get a ball if and only if

$$\sum_{w \in T_u} f(w) \geq |T_u| \quad \text{for all vertices } u.$$  

**Proof.** Since only the balls which start in a vertex of $T_u$ can end at $T_u$, the necessity is obvious. Conversely, since $\sum_{w \in T_u} f(w) > |T_u \setminus \{u\}|$, at least one car tried to park in $u$, hence any vertex $u$ is nonempty. Note that this doesn’t depend on the particular order of the cars, hence it also proves the permutation invariance of $T$-parking functions.

Theorem 1 can be viewed as a special case of the Marriage Theorem. However, because of the tree structure we have a much simpler form—instead of the usual $2^n$ inequalities we need only $n$. Theorem 1 allows us to describe $T$-parking distributions as non-negative integer solutions subject to certain linear constraints, since a distribution function $f$ is a $T$-parking distribution if and only if it satisfies $f(v) \geq |T_v|$ and $\sum_{v \in T} f(v) = |T|$. In general, if $f$ satisfies $f(v) \geq |T_v|$ and $\sum_{v \in T} f(v) = m$, then we have $m \geq n$ and there are $m - n$ balls dropping out of the root.

The permutation invariance also implies that the order of the terms in a $T$-parking function will not affect the number of failed parking attempts, or the number of bumps. For any vertex $v$ the number of cars that will move from that vertex towards the root is the difference between the number of cars having preference at the subtree $T_v$ and the number of vertices of $T_v$. In particular the order on the parking function will not affect how many bumps must happen at one vertex and hence will not affect the total number of bumps.
Let $f$ be a $T$-parking distribution. Denote by $\text{bump}(f)$ the number of bumps any $T$-parking function corresponding to the parking distribution $f$ would have. Then we have

$$\text{bump}(f) = \sum_v \left( \sum_{w \in T_v} f(w) - |T_v| \right).$$

For classical parking functions, the bump is called the total displacement, and the bump of car $c_i$ is the individual displacement. The values of total and individual displacements are of particular interest in problems related to hashing algorithms [3, 5, 6, 18].

### 3. Generating function for parking distributions

In this section we will produce a generating function for each rooted tree $T$, the constant term of which gives the number of $T$-parking distributions.

To illustrate the basic idea of the approach, consider the tree shown in Figure 3 consisting of taking two rooted trees $T_1$ and $T_2$ and combining them together by connecting their respective roots to a new root vertex $v$. One way to construct a $T$-parking distribution is to combine a $T_1$-parking distribution and a $T_2$-parking distribution and have one car prefer the root $v$. The other way to do this is to find a distribution on one of the trees with all vertices getting a parked car and one vehicle that exits together with a parking distribution on the other tree. In particular it is useful not only to know how many parking distributions there are on the subtrees, but also how many distributions there are where all vertices get a parked car and with one extra unparked car that exits.

![Figure 3: A tree $T$ with two subtrees off the root](image)

In general, we are interested in two different counts:

- $p_i(T)$: The number of $T$-parking distributions where each vertex gets a car parked and $i$ cars exit.
- $q_i(T)$: The bump $q$-analogue of the number of $T$-parking distributions where each vertex gets a car parked and $i$ cars exit. In formula, $q_i(T)$ is a polynomial of $q$ defined by

$$q_i(T) = \sum_f q^{\text{bump}(f)},$$

where $f$ ranges over all $T$-parking distributions counted by $p_i(T)$.

When it is clear from context we will simply use $p_i$ and $q_i$ respectively for these functions. Define their respective generating functions as

$$P_T(x) = \sum_{i \geq 0} p_i x^i, \quad Q_T(x) = \sum_{i \geq 0} q_i x^i.$$

The number and its bump $q$-analogue of the $T$-parking distributions are given by $P_T(0)$ and $Q_T(0)$, respectively.

**Observation 1.** If $T = \bullet$, then $P_T(x) = \frac{1}{1-x}$ and $Q_T(x) = \frac{1}{1-qx}$.

This is easy to see since a single vertex only has one parking distribution with $i$ cars exiting. Namely, the vertex repeated $i + 1$ times. So we have $p_i(\bullet) = 1$ and $q_i(\bullet) = q^i$. Putting these coefficients into the above generating functions and simplifying, we get the results.

Given rooted trees $T_1, T_2, \ldots, T_k$, let $T = T_1 \oplus T_2 \oplus \cdots \oplus T_k$ be the rooted tree formed by taking the union of the $T_i$ and adding a new root vertex $v$ that is connected to the roots of $T_i$. We say that $T$ is the direct sum of $T_1, T_2, \ldots, T_k$. 

Theorem 2. Let \( T = T_1 \oplus T_2 \oplus \cdots \oplus T_k \). Then
\[
P_T(x) = \frac{1}{x} \left( \prod_i P_{T_i}(x) - \prod_i P_{T_i}(0) \right),
\]
\[
Q_T(x) = \frac{1}{q^x} \left( \prod_i Q_{T_i}(qx) - \prod_i Q_{T_i}(0) \right).
\]

Proof. We prove the equation for \( Q_T(x) \); the one for \( P_T(x) \) is then obtained by considering \( q = 1 \). The result for \( Q_T(x) \) reduces to understanding the following equation:
\[
q_n(T) = q^n \sum_{j=0}^{n+1} \left( \sum_{(m_1, m_2, \ldots, m_k) \vdash j} \left( \prod_{i=1}^{k} q^{m_i} q_{m_i}(T_i) \right) \right),
\]
where \((m_1, m_2, \ldots, m_k) \vdash j (m_i \in \mathbb{N})\) indicates that we are summing over all ordered partitions of \( j \) into \( k \) parts.

To see this we group parking distributions by how many cars prefer the root. Since ultimately \( n \) cars will exit we can have anywhere from 0 to \( n+1 \) cars preferring the root. Suppose that \( n+1-j \) cars initially prefer the root. This means that we will need to have \( j \) more cars end up at the root and this can be done by having the cars come from the subtrees, in particular suppose that we will have \( m_i \) cars coming from \( T_i \), where \( \sum m_i = j \) (i.e., an ordered partition of \( j \)). The bumps can happen in two general locations: up among the trees \( T_i \), and at vertex \( v \). The term \( \prod_{i=1}^{k} q_{m_i}(T_i) \) takes care of counting the bumps that happen in the trees (i.e., the bumps in the exponents add when multiplying terms together). That leaves us to determine what happens at \( v \), but there are exactly \( n \) bumps at \( v \) which correspond to the \( n \) cars that leave (i.e., anything that is bumped at \( v \) exits un-parked); so to account for this we add the term \( q^n \) to the front.

Note that \( q_n(T) \) can be expressed as
\[
q_n(T) = \frac{1}{q} \sum_{j=0}^{n+1} q^{n+1-j} \left( \sum_{(m_1, m_2, \ldots, m_k) \vdash j} \left( \prod_{i=1}^{k} q^{m_i} q_{m_i}(T_i) \right) \right).
\]

From here we use three basic manipulations for generating functions (see [19]). The first is to note that
\[
\prod_i Q_{T_i}(qx) = \sum_{j=0}^{\infty} \left( \sum_{(m_1, m_2, \ldots, m_k) \vdash j} \left( \prod_{i=1}^{k} q^{m_i} q_{m_i}(T_i) \right) \right) x^j.
\]
The second is convolving this with \( 1/(1-qx) = 1 + qx + q^2x^2 + q^3x^3 + \cdots \) gives
\[
\frac{\prod_i Q_{T_i}(qx)}{1-qx} = \sum_{n=0}^{\infty} \left( \sum_{j=0}^{n} q^{n-j} \left( \sum_{(m_1, m_2, \ldots, m_k) \vdash j} \left( \prod_{i=1}^{k} q^{m_i} q_{m_i}(T_i) \right) \right) \right) x^n.
\]
Lastly, this is almost what we need, except that we need to shift the entries by 1 and divide out by a \( q \) (i.e., we are off by 1 in the exponent for \( q \)). This can be accomplished by subtracting off the constant term, i.e., \( \prod_i Q_{T_i}(0) \), and then dividing the result by \( qx \).

For reference we provide these two generating functions for all trees on two or three vertices in Table [1].

3.1. Computation of the generating functions

The main benefit of Theorem [2] is that this gives a way to compute the number of \( T \)-parking distributions for large trees without having to exhaustively generate all possible distributions.

Observation 2. If \( T \) is a tree on \( n \) vertices then \( P_T(x)(1-x)^n = M_T(x) \) where \( M_T(x) \) is a degree \( n-1 \) polynomial satisfying \( M_T(1) = 1 \).
Table 1: Generating functions for small trees.

<table>
<thead>
<tr>
<th>$T$</th>
<th>Generating functions</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="image" /></td>
<td>$P_T(x) = \frac{2 - x}{(1 - x)^2}$</td>
</tr>
<tr>
<td><img src="image2" alt="image" /></td>
<td>$Q_T(x) = \frac{(1 + q) - q^2x}{(1 - qx)(1 - q^2x)}$</td>
</tr>
<tr>
<td><img src="image3" alt="image" /></td>
<td>$P_T(x) = \frac{5 - 6x + 2x^2}{(1 - x)^3}$</td>
</tr>
<tr>
<td><img src="image4" alt="image" /></td>
<td>$Q_T(x) = \frac{(1 + 2q + q^2 + q^3) - (q^2 + 2q^3 + 2q^4 + q^5)x + (q^5 + q^6)x^2}{(1 - qx)(1 - q^2x)(1 - q^3x)}$</td>
</tr>
<tr>
<td><img src="image5" alt="image" /></td>
<td>$P_T(x) = \frac{3 - 3x + x^2}{(1 - x)^3}$</td>
</tr>
<tr>
<td><img src="image6" alt="image" /></td>
<td>$Q_T(x) = \frac{(1 + 2q) - (2q^2 + q^3)x + q^4x^2}{(1 - qx)(1 - q^2x)^2}$</td>
</tr>
</tbody>
</table>

This follows by noting it works for the case when $T$ is a single vertex and then verifying that the result holds when we build up the tree using the recursion from Theorem 2. Further we note that $M_T(0) = P_T(0)$ and so it suffices to compute $M_T(x)$ if we want to determine the number of parking distributions. Following Theorem 2 we have the following corollary.

**Corollary 3.** Let $T$ be a tree on $n$ vertices with root $v$ and assume $T = T_1 \oplus T_2 \oplus \cdots \oplus T_k$. Then the polynomial $M_T(x)$ can be computed by the recurrence

$$M_T(x) = \frac{\prod_i M_{T_i}(x) - (1 - x)^n \prod_i M_{T_i}(0)}{x}$$

and the initial condition that if $T = \bullet$, then $M_T = 1$.

The values for $M_T(x)$ for $|T| \leq 3$ can be found by Observation 1 and Table 1. In Table 2 we list the polynomials $M_T(x)$ for $|T| = 4, 5$. One thing to notice about all of these polynomials is that they are distinct. For trees up through 18 vertices it has been verified that all rooted trees have distinct polynomials.

Using the recursive approach we can compute $M_T(x)$ for any tree and also determine $M_T(x)$ for specific families.

**Example 2.** For $T = P_n$, the path with $n$ vertices with a root at a leaf, we have $M_{P_n}(x) = \sum_{k=0}^{n-1} h_{n,k} x^k$ where

$$h_{n,k} = (-1)^k \frac{(n-k)(n-k+1)}{(n+k)(n+k+1)} \binom{n+1}{k} C_n = \frac{(-1)^k}{n} \binom{2n}{n-1-k} \binom{n+k-1}{k}$$

and $C_n$ is the $n$th Catalan number.

The result in the preceding example can be established using the recursive definition of $M_T$ and some basic combinatorial identities; we leave the details to the interested reader.

**Remark 1.** We note that there is a method given $M_T(x)$ to reconstruct the tree(s) which it came from. Namely, given $M_T(x)$ let $(-1)^{n-1} c$ be the coefficient of $x^{n-1}$. We note that the product of the $M_{T_i}$ will have degree strictly less than $n$, and therefore $c$ is $\prod_i M_{T_i}(0)$. Then we have

$$F(x) := xM_T(x) + c(1 - x)^n = \prod_i M_{T_i}(x).$$

Now look at factorizations of $F(x)$ and recurse. Note that after we have factored we can determine how many leaves appended to the root there needed to be, i.e., how many of the $M_{T_i}(x) = 1$. It is possible that there are multiple legal factorizations and so this does not (yet) show that the polynomials are unique to their trees.
Exact formulas for the generating function $Q_T(x)$ are more complicated. Nevertheless, we observe that it can be expressed as a rational function of $q$ and $x$. To describe it we need some notation. Let $T$ be a rooted tree with the root vertex $v$. For any vertex $u$ in $T$, let $d(u)$, the depth of $u$, be the distance from $u$ to the root $v$. In particular $d(v) = 0$.

**Observation 3.** If $T$ is a rooted tree with $n$ vertices then

$$Q_T(x) \prod_{u \in T} (1 - q^{d(u)+1}) x = N_T(q, x)$$

where $N_T(q, x)$ is a polynomial in the variables $q, x$. In $N_T(q, x)$ the highest degree of $x$ is $n-1$. In addition, $N_T(1, x) = M_T(x)$ and $N_T(q, 0) = q_0(T)$.

We can again use Theorem 2 to get a recurrence of $N_T(q, x)$.

**Corollary 4.** If $T = \bullet$, then $N_T(q, x) = 1$. Otherwise, if $T$ is a tree with $n$ vertices and the root $v$, and $T = T_1 \oplus T_2 \oplus \cdots \oplus T_k$, then

$$N_T(q, x) = \prod_i N_{T_i}(q, x) - \prod_u (1 - q^{d(u)+1}) x \prod_i N_{T_i}(q, 0).$$

One notices that both $P_T(x)$ and $Q_T(x)$ are rational functions with special denominators. We will give a combinatorial explanation in the next subsection.

### 3.2 Positive of coefficients in $M_T(-x)$ and $N_T(q, -x)$

In this section we give a combinatorial interpretation for the rationality of $P_T(x)$ and $Q_T(x)$. As a corollary we prove that the coefficients of $M_T(-x)$ and $N_T(q, -x)$ are all positive.

Let us start with the classical case, i.e., where $T = P_n$ is the path with $n$ vertices. Assume that the vertices of $P_n$ are $v_1, \ldots, v_n$, where $v_i$ is the vertex labeled by $i$ and $v_n$ is the root. Let $C_{n,k}$ be the number of non-decreasing parking functions of length $n+1$ (i.e., $P_{n+1}$-parking functions) whose maximal element
is \( k + 1 \). Then for any distribution function \( f : \{v_1, \ldots, v_n\} \to \mathbb{N} \) such that every vertex of \( P_n \) gets a car parked and \( i \) cars exit, we can uniquely decompose \( f \) as the sum of two functions from \( \{v_1, \ldots, v_n\} \) to \( \mathbb{N} \) as follows: Let \( k \) be the index such that \( f(v_1) + f(v_2) + \cdots + f(v_k) < n \leq f(v_1) + f(v_2) + \cdots + f(v_{k+1}) \). Define

\[
f_1(v_i) = \begin{cases} f(v_i), & \text{if } i \leq k \\ n - \sum_{j=1}^{k} f(v_j), & \text{if } i = k + 1 \\ 0, & \text{if } i > k + 1. \end{cases}
\]

and \( f_2(v_i) = f(v_i) - f_1(v_i) \) for all \( i \). Then \( f_1 \) is a \( P_n \)-parking distribution whose maximal element is \( k + 1 \) and \( f_2 \) is a distribution of \( i \) identical balls over \( n - k \) distinguishable bins. It follows that

\[
P_{P_n}(x) = \sum_{k=0}^{n-1} \frac{C_{n-1,k}}{(1 - x)^{n-k}},
\]

and

\[
M_{P_n}(x) = \sum_{k=0}^{n-1} C_{n-1,k}(1 - x)^k,
\]

The last equation implies that all the coefficients of \( M_{P_n}(-x) \) are positive.

The triangle array \( \{C_{n,k} : n \geq 0, 0 \leq k \leq n\} \) is known as Catalan’s triangle, which is sequence A009766 in the Online Encyclopedia of Integer Sequences (OEIS) and has many combinatorial interpretations \[15\]. The coefficients \( h_{n,k} \) of \( M_{P_n}(x) \) are related to \( C_{n,k} \) via the equation

\[
(-1)^k h_{n,k} = \sum_{s=k}^{n-1} \binom{s}{k} C_{n-1,s}.
\]

The signless triangle array \( \{(-1)^k h_{n,k} : n \geq 1, 0 \leq k < n\} \) is called Borel’s triangle, which is sequence A234950 in the OEIS. (In the OEIS the row index of Borel’s triangle \( \{T_{n,k}\} \) starts at zero. Hence \( h_{n,k} = T_{n-1,k} \).) This sequence occurs as the Betti numbers of certain ideals in the polynomial ring \( K[x_1, \ldots, x_n] \) and arises in the enumeration of up-down patterns in permutations; see \[1] \[16\]. The first five rows of Borel’s and Catalan’s triangles are listed below.

<table>
<thead>
<tr>
<th>Borel’s triangle</th>
<th>Catalan triangle</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2 1</td>
<td>1 1</td>
</tr>
<tr>
<td>5 6 2</td>
<td>1 2 2</td>
</tr>
<tr>
<td>14 28 20 5</td>
<td>1 3 5 5</td>
</tr>
<tr>
<td>42 120 135 70 14</td>
<td>1 4 9 14 14</td>
</tr>
</tbody>
</table>

For the bump \( q \)-analogue, let \( C_{n-1,k}(q) \) be the bump-enumerator

\[
C_{n-1,k}(q) = \sum_{f} q^{\text{bump}(f)}
\]

where \( f \) ranges over all non-decreasing parking functions of length \( n \) with maximal element \( k + 1 \). Then the above decomposition of \( f \to (f_1, f_2) \) implies

\[
Q_{P_n}(x) = \sum_{k=0}^{n-1} \frac{C_{n-1,k}(q)}{(1 - qx)(1 - q^2x) \cdots (1 - q^{n-k}x)},
\]

and

\[
N_{P_n}(q, x) = Q_{P_n}(x) \prod_{i=1}^{n}(1 - q^i x) = \sum_{k=0}^{n-1} C_{n-1,k}(q)(1 - q^{n-k+1}x) \cdots (1 - q^n x).
\]

It follows from the last equation that all the coefficients of \( N_{P_n}(q, -x) \) are positive.
For a general tree, we want to find expressions of \( P_T(x) \) and \( Q_T(x) \) similar to (2) and (3). For a rooted tree of \( n \) vertices, label the vertices from 1 to \( n \) such that if \( u \) is a descendant of \( w \), then the label of \( u \) is less than the label of \( w \). (Hence the root must have label \( n \).) Fix this labeling throughout and let \( \nu_i \) be the vertex with label \( i \).

Define a parking configuration on \( T \) as a function from \( V(T) \) to \( \{0, 1\} \), where 0 means the vertex is empty and 1 means there is a car parked at the vertex. Let \( g : V(T) \to \mathbb{N} \) be a distribution function such that if we apply the parking process to this distribution, then each vertex of \( T \) gets a car. Define a parking distribution \( \rho_g \) as follows. Start with the empty parking configuration \( g^{(0)} \) on \( T \) and sequentially define \( n \) parking configurations \( g^{(1)}, g^{(2)}, \ldots, g^{(n)} \) on \( T \), where in each round some cars enter and look for parking spaces according to the usual parking rule. More precisely, from \( i = 1 \) to \( n \) we process the vertices \( v_1 \) to \( v_n \) one by one. In round \( i \), starting with the parking configuration \( g^{(i-1)} \), we put \( g(v_i) \) many new cars on the vertex \( v_i \) and apply the parking rules. Assume that \( k \) of these cars find some vertices of \( T \) to park and others exit. This results in a new parking configuration \( g^{(i)} \) of cars on the vertices of \( T \). Set \( \rho_g(i) = k \) and go to the next round.

The above procedure yields a \( T \)-parking distribution \( \rho_g \), which we call the core of \( g \). Clearly each distribution counted by \( p_i(T) \) has a core. If \( f \) itself is a \( T \)-parking distribution, then \( \rho_f = f \).

**Example 3.** Consider the tree given in Figure 4.

![Figure 4: A tree T with the root at 4](image)

Let \( g \) be the distribution \( g(v_1) = 1, g(v_2) = 2, g(v_3) = 2 \) and \( g(v_4) = 3 \). The parking configurations after rounds 1 through 4 are given in Figure 5, where a red vertex means there is a car parked at that vertex. The core \( \rho_g \) is the \( T \)-parking distribution given by \( \rho_g(v_1) = 1, \rho_g(v_2) = 2, \rho_g(v_3) = 1 \) and \( \rho_g(v_4) = 0 \).

![Figure 5: An example highlighting the core](image)

Conversely, given a \( T \)-parking distribution \( f \), we characterize all the distribution functions whose core is \( f \). For a vertex \( v_i \) of \( T \), we say that it is closed with respect to \( f \) if there is a \( T \)-parking distribution \( f' \) such that \( f(v_j) = f'(v_j) \) for \( 1 \leq j < i \) and \( f(v_i) < f'(v_i) \). Otherwise \( v_i \) is open with respect to \( f \). Note that the root is always an open vertex.

**Example 4.** Let \( T \) be the rooted tree \( P_1 \oplus P_1 \) with three vertices. The following table shows all the \( T \)-parking distributions and the open and closed vertices with respect to them.

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Open Vertices</th>
<th>Closed Vertices</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 )</td>
<td>( v_2, v_3 )</td>
<td>( v_1 )</td>
</tr>
<tr>
<td>( f_2 )</td>
<td>( v_1, v_3 )</td>
<td>( v_2 )</td>
</tr>
<tr>
<td>( f_3 )</td>
<td>( v_1, v_2 )</td>
<td>( v_3 )</td>
</tr>
</tbody>
</table>

It can be easily checked that for a distribution \( g \),

- the core of \( g \) is \( f_1 \) if and only if \( g(v_1) = g(v_2) = 1 \) and \( g(v_3) \geq 1 \);
- the core of \( g \) is \( f_2 \) if and only if \( g(v_1) = 1, g(v_2) \geq 2 \) and \( g(v_3) \geq 0 \);
- the core of \( g \) is \( f_3 \) if and only if \( g(v_1) \geq 2, g(v_2) \geq 1 \) and \( g(v_3) \geq 0 \).
Theorem 5. Let \( f \) be a \( T \)-parking distribution. Then for a distribution function \( g \), the core of \( g \) is \( f \) if and only if the following linear constraints hold:

\[
\begin{align*}
g(u) &= f(u) \quad \text{for any closed vertex } u \text{ with respect to } f, \\
g(u) &\geq f(u) \quad \text{for any open vertex } u \text{ with respect to } f.
\end{align*}
\]

Proof. Assume \( \rho g = f \). We compare the procedures of defining the cores for \( g \) and \( f \). We show by induction that at each round, \( g^{(i)} = f^{(i)} \) and \( g(v_i) \geq f(v_i) \), where the equality holds if \( v_i \) is closed. The claim is true for the base case \( i = 0 \), since \( g^{(0)} = f^{(0)} = \emptyset \) by definition.

Assume the claim is true for \( i - 1 \geq 0 \), we shall show it for \( i \). At the beginning of round \( i \), the parking configurations \( g^{(i-1)} \) and \( f^{(i-1)} \) are the same. When one adds \( g(v_i) \) new cars to the vertex \( v_i \), exactly \( f(v_i) \) of them find parking spaces, which must be the first \( f(v_i) \) available spaces on the path from \( v_i \) to the root. Hence \( g(v_i) \geq f(v_i) \) and \( g^{(i)} = f^{(i)} \). If \( v_i \) is a closed vertex, then on the path from \( v_i \) to the root, there are still empty spaces. Thus we must have \( g(v_i) = f(v_i) \), otherwise \( g(v_i) > f(v_i) \) then at least \( f(v_i) + 1 \) of the new cars entered at \( v_i \) can park, which is a contradiction.

The converse is similar. \( \square \)

Let \( \text{Open}(f) \) and \( \text{Closed}(f) \) be the set of open and closed vertices of \( T \) with respect to \( f \) and \( \text{op}(f) = |\text{Open}(f)| \), \( c(f) = |\text{Closed}(f)| \). Note that \( \text{Open}(f) \neq \emptyset \) since the root is always open. Then Theorem 5 implies that

\[
P_T(x) = \sum_f \frac{1}{(1-x)^{\text{op}(f)}} = \sum_{k=1}^{n} \frac{d_k(T)}{(1-x)^k},
\]

where \( f \) ranges over all \( T \)-parking distributions and \( d_k(T) \) is the number of \( T \)-parking distributions with \( k \) open vertices. Similarly,

\[
Q_T(x) = \sum_f \prod_{u \in \text{Open}(f)} \frac{q^{\text{bump}(f)}}{(1-q^{d(u)+1})},
\]

where \( f \) ranges over all \( T \)-parking distributions. It follows that the coefficients of \( M_T(-x) \) and \( N_T(q,-x) \) are all positive.

3.3. Log-concavity of coefficients of \( M_T(-x) \)

Examination of the entries in Table 2 suggests that the coefficients of \( M_T(-x) \) not only are positive, but also form a log-concave sequence, i.e., satisfy \( a_i^2 \geq a_{i-1}a_{i+1} \) for any three consecutive terms. This holds in general and the following approach was communicated to us by Richard Stong [17].

Lemma 6. Let \( q(t) \) be a polynomial of degree at most \( n - 1 \) with non-negative integer coefficients and \( q(0) \neq 0 \). Then

\[
Q(t) = \frac{t^{n+1}q(1) - q(t)}{t - 1}
\]

is a polynomial of degree \( n \) with non-decreasing positive integer coefficients and \( P(x) = Q(x + 1) \) is a polynomial of degree \( n \) with (strictly) log-concave positive integer coefficients.
Proof. Let \( q(t) = \sum_{k=0}^{n-1} q_k t^k \) and take \( q_n = 0 \) by convention. Then we compute
\[
Q(t) = \sum_{k=0}^{n-1} q_k \frac{t^{n+1} - t^k}{t-1} = \sum_{k=0}^{n-1} q_k (t^k + t^{k+1} + \cdots + t^n) = \sum_{d=0}^{n} \left( \sum_{k=0}^{d} q_k \right) t^d.
\]
From this and the hypothesis that \( q_0 = q(0) > 0 \), it is obvious that the coefficients of \( Q(t) \) are non-decreasing positive integers.

From the above formula for \( Q(t) \) we have
\[
P(x) = Q(x+1) = \sum_{k=0}^{n-1} q_k \frac{(x+1)^{n+1} - (x+1)^k}{x} = \sum_{k=0}^{n-1} q_k \sum_{d=0}^{n} \left( \begin{array}{c} n+1 \\ d+1 \end{array} \right) \left( \begin{array}{c} k \\ d+1 \end{array} \right) x^d
\]
\[
= \sum_{d=0}^{n} \left( \sum_{k=0}^{n-1} q_k \left( \begin{array}{c} n+1 \\ d+1 \end{array} \right) \left( \begin{array}{c} k \\ d+1 \end{array} \right) \right) x^d.
\]
From this it is obvious that the \( P(x) = \sum_{k=0}^{n} p_k x^k \) is a polynomial with positive integer coefficients. Proving that the coefficients are strictly log-concave amounts to proving the inequality \( p_r^2 - p_{r-1}p_{r+1} > 0 \) for \( r = 1, \ldots, n-1 \). Since each \( p_k \) is a linear function of the coefficients \( q_k \) of \( q(t) \), the right hand side of the inequality is quadratic in these coefficients. We will prove the inequality by showing that in fact the coefficient of \( q_k q_m \) is positive for all \( 0 \leq k, m \leq n-1 \). This amounts to the inequality
\[
2 \left( \begin{array}{c} n+1 \\ r+1 \end{array} \right) \left( \begin{array}{c} n+1 \\ r+1 \end{array} \right) \left( \begin{array}{c} n+1 \\ r+1 \end{array} \right) \left( \begin{array}{c} m \\ r+1 \end{array} \right) - \left( \begin{array}{c} n+1 \\ r \end{array} \right) \left( \begin{array}{c} n+1 \\ r \end{array} \right) \left( \begin{array}{c} m \\ r \end{array} \right) - \left( \begin{array}{c} n+1 \\ r+2 \end{array} \right) \left( \begin{array}{c} n+1 \\ r+2 \end{array} \right) \left( \begin{array}{c} m \\ r+2 \end{array} \right) - \left( \begin{array}{c} n+1 \\ r+2 \end{array} \right) \left( \begin{array}{c} n+1 \\ r+2 \end{array} \right) \left( \begin{array}{c} m \\ r \end{array} \right) > 0.
\]
Using the identity \( \binom{n}{r+1} = \binom{n}{r} \binom{n-r-1}{r} \) and rearranging we can rewrite this as
\[
2(n+1-r)(n+2) \frac{(n+1)}{(r+2)(r+1)^2} \left( \begin{array}{c} n+1 \\ r \end{array} \right) \left( \begin{array}{c} n+1 \\ r \end{array} \right) \left( \begin{array}{c} m \\ r \end{array} \right) + \frac{(n+1-m)(r(n-m)+(3n+4-m))}{(r+2)(r+1)^2} \left( \begin{array}{c} m \\ r \end{array} \right) \left( \begin{array}{c} n+1 \\ r \end{array} \right) - \left( \begin{array}{c} k \\ r \end{array} \right) + \frac{(n+1-k)(r(n-k)+(3n+4-k))}{(r+2)(r+1)^2} \left( \begin{array}{c} k \\ r \end{array} \right) \left( \begin{array}{c} n+1 \\ r \end{array} \right) - \left( \begin{array}{c} m \\ r \end{array} \right) + \frac{2(n+1-k)(n+1-m)}{(r+1)^2} \left( \begin{array}{c} k \\ r \end{array} \right) \left( \begin{array}{c} m \\ r \end{array} \right) > 0.
\]
Since each term of this expression is clearly non-negative and the first term is strictly positive, the result follows.

\[\Box\]

Theorem 7. For any tree \( T \), the coefficients of \( M_T(-x) \) are positive and log-concave.

Proof. We proceed by induction on the number of vertices and work with \( K_T(t) = M_T(1-t) \). For the base case we have \( M_{\emptyset}(-x) = 1 \).

Now suppose that for all rooted trees \( T \) with at most \( n \) vertices that \( K_T(t) \) has degree \(|T| - 1\), non-decreasing positive integer coefficients, and \( K_T(0) > 0 \). Now let \( \hat{T} \) be a rooted tree on \( n + 1 \) vertices with
root \(v\) and that removing the root leaves trees \(T_1, \ldots, T_k\). Then
\[
K_{\hat{T}}(t) = M_{\hat{T}}(1 - t) = \prod_i M_{T_i}(1 - t) - \frac{t^{n+1} \prod_i M_{T_i}(0)}{1 - t} = \prod_i K_{T_i}(t) - \frac{t^{n+1} \prod_i K_{T_i}(1)}{1 - t} = \frac{t^{n+1} \prod_i K_{T_i}(1) - \prod_i K_{T_i}(t)}{t - 1}.
\]

Now note that by the induction hypothesis, \(q(t) = \prod_i K_{T_i}(t)\) is a polynomial of degree at most \(n - 1\) (more precisely the degree is \(n\) minus the number of subtrees connected to the root), with non-negative integer coefficients, and \(q(0) > 0\). Therefore applying Lemma 6 we conclude that \(K_{\hat{T}}(t)\) has degree \(n\), non-decreasing positive integer coefficients. Further we have \(K_{\hat{T}}(t + 1) = M_{\hat{T}}(-t)\) has (strictly) log-concave positive integer coefficients, establishing the result.

\[\square\]

4. Caterpillars and lattice walks

The purpose of this section is to enumerate the number of \(T\)-parking distributions for special families of trees. We will present explicit formulas for \(p_0(T)\) for some families. A basic approach is to use the recurrence proved in Theorem 2 and then read the constant coefficient. The formula also calls for the values of \(p_1(T)\), the number of distributions on \(T\) where each vertex gets a car parked and one car exits. In other words, given a rooted tree \(T\), let \(\hat{T}\) be the rooted tree obtained from \(T\) by connecting a new vertex \(\hat{v}\) to the root of \(T\), and reset the root to be at \(\hat{v}\). Then \(p_1(T)\) is the number of parking distributions on \(\hat{T}\) such that there is no car preferring \(\hat{v}\), i.e., \(p_1(T) = p_0(\hat{T}) - p_0(T)\). We have the following general theorem.

**Theorem 8.** Let \(T = T_1 \oplus T_2 \oplus \cdots \oplus T_k\). Then
\[
p_0(T) = \left(1 + \sum_{i=1}^k \frac{p_1(T_i)}{p_0(T_i)}\right) \prod_{i=1}^k p_0(T_i).
\]

**Proof.** The formula can be obtained from the recurrence in Theorem 2. Here we give a direct combinatorial argument. Let \(f\) be a \(T\)-parking distribution. Clearly for the root vertex \(v\), we must have \(f(v) \leq 1\).

If \(f(v) = 1\), the restriction of \(f\) on each branch \(T_i\) must be a parking distribution of \(T_i\). There are \(\prod_{i=1}^k p_0(T_i)\) such parking distributions.

If \(f(v) = 0\), then the car parking at the spot \(v\) at the end of the parking process must come from a branch \(T_i\) for \(i \in [k]\). This implies that \(f\) is a parking distribution when restricted to \(T_j\) for \(j \neq i\); while restricted to \(T_i\), each vertex of \(T_i\) gets a car and exactly one car exits \(T_i\). There are \(p_1(T_i) \prod_{j \neq i} p_0(T_j)\) such parking distributions.

From Theorem 1 we know that \(p_1(T_i)\) equals the number of non-negative integer solutions of the system
\[
\sum_{v \in V(T)} x_v = n + i
\]
\[
\sum_{w \in T_u} x_w \geq |T_u| \quad \text{for all } u \in V(T).
\]

For certain families of rooted trees, we can count the number of non-negative integer solutions of the above systems by using lattice paths in plane. Take \(T = P_n\) as an example. For \(i = 0\) the equations (4) and (5) become \(x_1 + x_2 + \cdots + x_n = n\) and \(x_1 + \cdots + x_k \geq k\) for all \(k = 1, 2, \ldots, n\). It is well-known that the number of non-negative integer solutions of the above system is given by the \(n\)-th Catalan number \(C_n\).

For \(p_1(P_n)\), the equations (4) and (5) become \(x_1 + x_2 + \cdots + x_n = n + i\) and \(x_1 + \cdots + x_k \geq k\) for all \(k = 1, 2, \ldots, n\). Each non-negative integer solution of this system can be represented by a lattice path in the Euclidean plane for each solution of the equation \(x_1 + \cdots + x_n = n + i\), we start from the origin and represent \(x_i\) by a horizontal step \((x_i, 0)\) and each addition sign by a vertical step \((0, 1)\). Then we obtain a lattice paths from \((0,0)\) to \((n+i, n-1)\). The inequalities \(x_1 + \cdots + x_k \geq k\) for all \(k\) imply that the path
is sub-diagonal, i.e., never goes above the diagonal \( y = x \). Using the result that the number of sub-diagonal lattice paths from \((0, 0)\) to \((a, b)\) for integers \(a \geq b\) is \(\frac{a-b+1}{a+b+1}(a+b+1)\), (see [12]), we obtain that

\[
p_i(P_n) = \frac{i + 2}{2n + i} \binom{2n + i}{n - 1}.
\]

In particular,

\[
p_1(P_n) = \frac{3}{2n + 1} \binom{2n + 1}{n - 1} = \frac{3n}{n + 2} C_n.
\]

Applying Theorem 8 and the above formulas, we obtain the formula of \(p_0(T)\) when \(T\) is a superstar or a rooted tree of depth no more than two.

**Corollary 9.** Let \(m_1, \ldots, m_k\) be positive integers and \(T = P_{m_1} \oplus \cdots \oplus P_{m_k}\) be a superstar. Then

\[
p_0(T) = \left(1 + \sum_{i=1}^k \frac{3m_i}{m_i + 2}\right) \prod_{i=1}^k C_{m_i}.
\]

In particular, for \(\text{Star}_n\), the star with \(n\) vertices, \(p_0(\text{Star}_n) = n\).

**Corollary 10.** Let \(T\) be a rooted tree of depth no more than two, that is, the distance between any vertex \(u\) and the root \(v\) is no more than 2. Then \(T\) is a direct sum of stars, i.e., \(T = \text{Star}_{m_1} \oplus \text{Star}_{m_2} \oplus \cdots \oplus \text{Star}_{m_k}\) for some positive integers \(m_1, \ldots, m_k\). We have

\[
p_0(T) = \left(1 + \sum_{i=1}^k \frac{m_i + 1}{2}\right) \prod_{i=1}^k m_i.
\]

**Proof.** We just need to compute \(p_1(\text{Star}_n)\). Let \(u\) be the vertex of \(\text{Star}_n\) with coordinate \(u\) and \(v\) the root. Clearly \(f(v) \leq 2\).

If \(f(v) = 2\), then there is only one possible distribution, namely, \(f(u) = 1\) for all \(u \neq v\).

If \(f(v) = 1\), then there are \(n - 1\) distributions, corresponding to integer solutions of the equation \(x_1 + x_2 + \cdots + x_{n-1} = n\) and \(x_i \geq 1\).

If \(f(v) = 0\), then there are \(\binom{n}{2}\) such parking distributions, corresponding to integer solutions of the equation \(x_1 + x_2 + \cdots + x_{n-1} = n + 1\) and \(x_i \geq 1\).

Combining the above three cases, we obtain \(p_1(\text{Star}_n) = \frac{n(n+1)}{2} = \frac{n+1}{2} p_0(\text{Star}_n)\), which leads to (6). \(\square\)

Next we turn to another family of trees, namely, caterpillars, for which the number \(p_i(T)\) can be expressed as the number of lattice paths with certain right boundaries.

Let \(a_1, a_2, \ldots, a_k\) be positive integers with \(a_1 \geq 2\). A caterpillar \(T = \text{Cat}(a_1, a_2, \ldots, a_k)\) is a tree which contains a path \(v_1 \cdots v_k\) where \(v_k\) is the root, and each \(v_i\) is connected to an additional \(a_i - 1\) leaves, for \(1 \leq i \leq k\). (Hence the farthest vertex from the root has depth \(k\).) See Figure 6 for the example \(\text{Cat}(4, 2, 4, 2)\) where \(k = 4\) and the path \(v_1 - v_2 - v_3 - v_4\) is 4 - 6 - 10 - 12 with vertex 12 being the root. Let \(N = \sum_{i=1}^k a_i\), which is the number of vertices of \(\text{Cat}(a_1, \ldots, a_k)\).

Our main tool is the enumeration of lattice paths in the plane with a given right boundary. Let us review the basic notation in lattice path counting. Let \(s\) be a non-decreasing sequence with positive integer terms \(s_0, s_1, s_2, \ldots\), thought of as a strict right boundary. A lattice path from \((0, 0)\) to \((x, n)\) is represented by the sequence \((x_0, x_1, \ldots, x_{n-1})\) where \((x_i, i)\) is the rightmost point in the lattice path with \(y\)-coordinate \(i\). Such a path is an \(s\)-lattice path if \(x_i < s_i\) for \(0 \leq i \leq n - 1\). Figure 7 shows the lattice path \((1, 1, 6, 8)\) from \((0, 0)\) to \((11, 4)\), which stays on the left of the boundary \(s = (4, 6, 10, 12, \ldots)\), indicated by the black dots. (In other words, the allowed lattice path cannot touch any points in the shaded area.)

If \(x \geq s_{n-1} - 1\), then the number of \(s\)-lattice paths from \((0, 0)\) to \((x, n)\) does not depend on \(x\). Let \(\text{LP}_n(s)\) be this common number. For example, when \(s = 1, 2, 3, \ldots\), then \(\text{LP}_n(s)\) is just the \(n\)-th Catalan number \(C_n\). By convention, we always let \(\text{LP}_0(s) = 1\).

**Theorem 11.** Let \(T = \text{Cat}(a_1, \ldots, a_k)\) be a caterpillar. Then

\[
p_0(T) = \text{LP}_k(a_1, a_2, \ldots, a_1 + \cdots + a_k),
\]
Figure 6: The caterpillar Cat(4,2,2)

Figure 7: The lattice path (1,1,6,8) with boundary s = (4,6,10,12,...)

Proof. We establish a bijection between $T$-parking distributions and certain lattice paths. First, index the vertices of $T$ from 1 to $N$ as follows. The leaves connected to vertex $v_1$ are indexed from 1 to $a_1 - 1$, $v_1$ is indexed by $a_1$; then the leaves connected to $v_2$ are indexed by $a_1 + 1, \ldots, a_1 + a_2 - 1$ and $v_2$ is indexed by $a_1 + a_2$. In general, $v_i$ is indexed by $a_1 + \cdots + a_i$ and the leaves connected to $v_i$ are indexed by $a_1 + \cdots + a_i - 1$ to $a_1 + \cdots + a_i - 1$. The root is labeled by $N$. The labeling for Cat(4,2,4,2) is shown in Figure 6.

In $T$ there are $k$ non-leaf vertices (including the root) and $N - k$ leaves. For each leaf $j$, any parking distribution $f$ must have $f(j) \geq 1$. Let

$$x_j \begin{cases} f(j) - 1 & \text{if } j \text{ is a leaf,} \\ f(j) & \text{otherwise.} \end{cases}$$

Then $f$ is a $T$-parking distribution if and only if

$$x_1 + x_2 + \cdots + x_N = k$$  \hspace{1cm} (7)$$

$$x_1 + x_2 + \cdots + x_{a_1 + \cdots + a_i} \geq i \quad \text{for } i = 1, 2, \ldots, k.$$  \hspace{1cm} (8)

For each non-negative integer solution of (7), we map it to a lattice path starting from (0,0) by replacing each “+” by a horizontal step (1,0) and replacing integer $x_j$ by a vertical step of length $x_j$. Then the non-negative integer solutions of (7) are in one-to-one correspondence to lattice paths in the plane from the origin to the point $(N - 1,k)$. The constraint (8) implies that the lattice path is strictly on the left of the boundary $(a_1, a_1 + a_2, \ldots, a_1 + \cdots + a_k)$. For example, for $T = \text{Cat}(4,2,4,2)$ in Figure 6, the corresponding
linear system defined by (7) and (8) is
\[
\begin{align*}
    x_1 + x_2 + \cdots + x_{12} &= 4 \\
    x_1 + x_2 + \cdots + x_4 &\geq 1 \\
    x_1 + x_2 + \cdots + x_6 &\geq 2 \\
    x_1 + x_2 + \cdots + x_{10} &\geq 3.
\end{align*}
\]

As an example consider the following solution of the above system, \(x_2 = 2, x_7 = x_9 = 1\) and all other \(x_i = 0\). It corresponds to the lattice path \(ENNEEEEENEENEE\) from \((0,0)\) to \((11,4)\), as depicted by the lattice path in Figure 7. The right boundary of this path is \((1,1,1,6,8)\), which is less than \(s = (4,6,10,12)\).

For \(p_i(T)\), the only difference is that (7) becomes
\[
\sum_{j=1}^{N} x_j = k + i.
\]

Hence the solutions correspond to lattice paths from the origin to \((N-1,k+i)\) with the strict right boundary \((a_1, a_1 + a_2, \ldots, a_1 + \cdots + a_k)\) on the first \(k\) rows, or equivalently, the right boundary \((a_1, a_1 + a_2, \ldots, N, N, \ldots, N)\) of length \(k+i\).

Lattice paths are a classical subject in combinatorial theory, whose enumeration has been systematically approached by symbolic computation and umbral calculus, see, for example, the book by Mohanty [12] and extensive work done by Niederhausen and collaborators, e.g. [13, 14, 8]. In particular, it is known that lattice paths within general boundaries can be computed by a determinant formula (see [12, p.32]).

**Theorem 12.** Let \((b_1, b_2, \ldots, b_k)\) be a sequence of non-decreasing integers. Then
\[
\text{LP}_k(b_1, b_2, \ldots, b_k) = \det \left[ \begin{array}{c} b_r \\ s - r + 1 \end{array} \right]_{1 \leq r,s \leq k}.
\]

Combining with Theorem 11 we have the following result.

**Corollary 13.** For a caterpillar \(T = \text{Cat}(a_1, a_2, \ldots, a_k)\), the number of \(T\)-parking distribution is given by
\[
p_0(T) = \det \left[ \begin{array}{c} a_1 + \cdots + a_r \\ s - r + 1 \end{array} \right]_{1 \leq r,s \leq k}.
\]

In addition, by setting \(a_{k+1} = a_{k+2} = \cdots = a_{k+i} = 0\) we can express \(p_i(T)\) by a similar determinant of a matrix of size \((k+i) \times (k+i)\).
\[
p_i(T) = \det \left[ \begin{array}{c} a_1 + \cdots + a_r \\ s - r + 1 \end{array} \right]_{1 \leq r,s \leq k+i}.
\]

Let us look at some special cases for \(T = \text{Cat}(a_1, a_2, \ldots, a_k)\).

1. \(k = 1\). Then \(\text{Cat}(n)\) is a star with \(n\) vertices. The determinantal formula again gives \(p_0(\text{Star}_n) = n\).
2. \(k = 2\). From Corollary 13
\[
p_0(T) = \det \left( \begin{array}{cc} a_1 & \binom{a_1}{2} \\ 1 & a_1 + a_2 \end{array} \right) = a_1 \left( a_2 + \frac{a_1 - 1}{2} \right).
\]

On the other hand, \(T\) can be viewed as the direct sum of \(\text{Star}_{a_1}\) and \(a_2 - 1\) isolated points (which are stars of one vertex). It is easy to check that the formula above agrees with (6).
3. We say that the caterpillar is $t$-regular if $a_1 - 1 = a_2 = \cdots = a_k = t$. In this case $p_0(T) = \text{LP}_k(1 + t, 1 + 2t, \ldots, 1 + kt)$, the number of lattice paths with a linear boundary. It is known (see [10]) that

$$\text{LP}_n(r, r + \mu, r + 2\mu, \ldots, r + (n - 1)\mu) = \frac{r}{r + n(\mu + 1)} \binom{r + n(\mu + 1)}{n}. \quad (11)$$

Substituting $r = 1 + t$, $n = k$ and $\mu = t$ into (11), we obtain

$$p_0(T) = \frac{1}{1 + t + k(t + 1)} \binom{1 + t + k(t + 1)}{k} = \frac{1}{(1 + t)(k + 1) + 1} \binom{(1 + t)(1 + k) + 1}{k + 1},$$

which is the $(k + 1)$-st term in the $(1 + t)$-Fuss-Catalan sequence, a natural generalization of the classical Catalan numbers. Explicitly, the $n$-th term of the $r$-Fuss-Catalan number is given by $\frac{1}{r^n + 1} \binom{r^n + 1}{n}$, and the classical Catalan numbers correspond to the case $r = 2$.

The numbers of lattice paths with a given right boundary $s$ satisfy an Appell equation (see [10, 11]), which is a variation of the ordinary generating function:

$$\sum_{n=0}^{\infty} \text{LP}_n(s)x^n(1 - x)^s = 1. \quad (12)$$

Let $s_i = a_1 + \cdots + a_{i+1}$ for $i = 0, \ldots, k - 2$, and $s_j = N = \sum_{i=1}^{k} a_i$ for $j \geq k - 1$. Combining Theorem 11 and 12 we obtain a formula for the generating function $P_T(x) = \sum_{i \geq 0} p_i(T)x^i$ when $T$ is a caterpillar.

**Theorem 14.** Let $T = \text{Cat}(a_1, a_2, \ldots, a_k)$ be a caterpillar and $N = \sum_{i=1}^{k} a_i$. Then

$$P_T(x) = \frac{1}{(1 - x)^N} \left(1 - \sum_{i=0}^{k-1} x^i(1 - x)^{s_i} \cdot \text{LP}_i(s_0, s_1, \ldots, s_{i-1})\right).$$

Consequently, if $M_T(x) = (1 - x)^N P_T(x)$, then

$$M_T(x) = \frac{1}{x^k} \left(1 - \sum_{i=0}^{k-1} x^i(1 - x)^{s_i} \cdot \text{LP}_i(s_0, s_1, \ldots, s_{i-1})\right).$$

(From 12 it can be seen that the polynomial in the parenthesis is a multiple of $x^k$.)

Let us apply Theorem 14 to $P_n$ for $n \geq 2$, which is the caterpillar $\text{Cat}(a_1, \ldots, a_{n-1})$ with $a_1 = 2$ and $a_2 = \cdots = a_{n-1} = 1$. The corresponding lattice paths has boundary $s = (2, 3, \ldots, n - 1, n, n, n, \ldots)$. Using Formula 11 we have $\text{LP}_i(s) = C_{i+1}$ for $i = 1, 2, \ldots, n - 1$. Hence

$$M_{P_n}(x) = \frac{1}{x^{n+1}} \left(1 - \sum_{i=0}^{n-2} C_{i+1}x^i(1 - x)^{i+2}\right). \quad (13)$$

The formula inside the parenthesis is a multiple of $x^{n-1}$, which can be checked using the equation $\sum_{i \geq 0} C_i(x(1 - x))^i = \frac{1}{1 - x}$. Consequently, for $0 < k < n - 1$, if $[x^k]h(x)$ is the coefficient of $x^k$ in a polynomial $h(x)$, then

$$[x^k] \sum_{i=0}^{n-2} C_{i+1}x^i(1 - x)^{i+2} = 0.$$

It leads to an interesting binomial identity

$$\sum_{i=0}^{k} (-1)^{k+i} C_{i+1} \binom{i + 2}{k - i} = 0,$$

or equivalently, for $k \geq 1$,

$$\sum_{i=0}^{k} (-1)^i \binom{k + 1}{k - i} \binom{2i + 2}{k} = 0,$$

which we leave as an exercise.
5. Concluding remarks and open questions

By utilizing the uniqueness of paths between vertices in a tree we have a natural generalization of parking functions to rooted trees. We note that for other families of graphs we lose this uniqueness. In particular this approach will not immediately generalize to all graphs without adding additional structure.

There are many different directions to go with this approach. There do exist distinct trees with the same number of parking distributions (see Table 2 for an example), are there easy ways to characterize some of these pairs? Does there exist a pair of distinct trees with the same generating function $P_T$ or $Q_T$? We comment that up through 18 vertices no such pair exists.

In Section 4 we looked at a special family and tied the result to lattice walks. Our calculation shows that the bump of a caterpillar is related to the area of the lattice path in a complicated way. Can this lead to a closed formula for the bump q-analogue? Are there other families where the parking distributions can readily be determined? In a related question, can we give other combinatorial interpretations to the number of parking distributions on trees with more structure?

We have focused on parking distributions in this paper, but one can also consider counting the ordered variation as well. We note without proof that for a given tree $T$, if we let $r_i$ be the number of (ordered) parking preferences which lead to all vertices filled and $i$ cars exiting and let $R_T(x) = \sum_{i=0}^{\infty} r_i \frac{x^i}{(i+|T|)!}$, then for the tree on a single vertex $R_T(x) = (e^x - 1)/x$ and using the same notation of Theorem 2 we have $R_T(x) = \frac{1}{x} (e^x \prod_i R_T(x) - \prod_i R_T(0))$.

We look forward to seeing more theory on parking distributions developed in future work.

Acknowledgements

The authors would like to thank the two anonymous referees for their careful comments and helpful suggestions. The third author was partially supported by NSF grant DMS-1161414.


