Edited by Miklós Bóna

Handbook of Enumerative Combinatorics

Catherine H. YanDepartment of Mathematics, Texas A&M University, College Station, TX

Chapter 1

Parking Functions

CONTENTS

1.1 Introduction

The notion of parking functions was introduced by Konheim and Weiss [53] as a colorful way to describe their work on computer storage. The parking problem can be stated as follows. There are n cars $C_1,\ldots,C_n$ that want to park on a one-way street with ordered parking spaces $0,1,\ldots,n-1$. Each car C_i has a preferred space a_i. The cars enter the street one at a time in the order $C_1,C_2,\ldots,C_n$. A car tries to park in its preferred space. If that space is occupied, then it parks in the next available space. If there is no space then the car leaves the street. The sequence $(a_1,\ldots,a_n)$ is called a *parking function of length n* if all the cars can park, i.e., no car leaves the street.

It is easy to see that a sequence $(a_1,\ldots,a_n)$ is a parking function if and only if it has at least i terms less than i, for each $1 \le i \le n$. Let $a_{(1)} \le a_{(2)} \le \cdots \le a_{(n)}$ be the non-decreasing rearrangement of terms a_i's, where $a_{(i)}$ is called the i-th order statistics of $(a_1,\ldots,a_n)$. Then the sequence $(a_1,\ldots,a_n)$ is a parking function if and only if $0 \le a_{(i)} < i$. * Another equivalent definition is that a sequence $(a_1,\ldots,a_n)$ is a parking function if and only if there is a permutation $\sigma \in \mathfrak{S}_n$ such that $0 \le a_{\sigma(i)} < i$.

The number of parking functions of length n is $(n+1)^{n-1}$, a result obtained in the very first paper [53] on the subject by analytic method. A simple and elegant proof was given by Pollak, (see [71]), which can be described as follows: Add an additional space n and arrange all $n+1$ spaces clockwise in a circle. Again assume that n cars enter the street one at a time, each with a preferred space $a_i \in \{0,1,\ldots,n\}$. Preference $a_i = n$ is treated like any other preference: if space n is occupied, car C_i moves clockwise to the first unoccupied space. Every sequence of preferences leaves one space unoccupied, and because of symmetry the number of sequences leaving a given space, say k, unoccupied is the same for every k, $0 \le k \le n$. Hence the number with $k = n$, which is the number of parking functions, is $\frac{1}{n+1}$ of the total number of preference sequences. This gives $\dfrac{(n+1)^n}{n+1} = (n+1)^{n-1}$.

It is immediately noticed that $(n+1)^{n-1}$ is the number of labeled trees on $n+1$ vertices, by the famous Cayley's formula [18]. Many bijections between the set of parking functions and the set of labeled trees are constructed. They reveal deep connections between parking functions and other combinatorial structures, and lead to various generalizations and applications in different fields, notably in algebra, interpolation theory, probability and statistics, representation theory, and geometry. In this chapter we survey the basic results and developments on the combinatorics theory of parking functions in the last 20 years.

Notation. We write $[n]$ for the set $\{1,2,\ldots,n\}$, and $[n]_0$ for the set $\{0,1,\ldots,n\}$. We will often think of a function $f : [n] \to [n]$ as the sequence of its values $f(1), f(2),\ldots, f(n)$. If (h_i) is a sequence, we will use the boldface letter $\mathbf{h}$ to represent it. The letter $\mathbf{a}$ is reserved for a parking function of length n. We denote by $\mathscr{PK}_n$ the set of parking functions of length n,

A labeled tree on $[n]_0$ is a connected graph on the vertex set $[n]_0$ with no cycles. It is equivalent to a labeled rooted forest on $[n]$ obtained from the tree by deleting vertex 0 and replacing edges connecting 0 to vertex i by a root at i. A labeled rooted forest on $[n]$ is also called an *acyclic function* on $[n]$. We denote by $\mathscr{T}_{n+1}$ the set of labeled trees with the vertex set $[n]_0$.

*In literature, some people assume that a_i are positive integers and use the condition that $1 \le a_{(i)} \le i$. Through out this chapter we always allow the value of 0 and require that $0 \le a_{(i)} < i$.

1.2 Parking Functions and Labeled Trees

There are many bijections between the set of parking functions of length n and the set of labeled trees on $[n]_0$, for example, see [72, 71, 51, 30, 31, 56, 65]. Here we introduce some elegant constructions, each of which reveals an intrinsic property of parking functions.

1.2.1 Labeled trees with Prüfer code

A Prüfer code, first introduced by Prüfer [70] to prove Cayley's formula, associates with each labeled tree with n vertices a unique sequence of length $n-2$. Given a tree $T \in \mathscr{T}_{n+1}$, its Prüfer code is generated by iteratively removing vertices from the tree until only two vertices remain. Specifically, at step i, one removes the leaf with the smallest label and sets the i-th element of the Prüfer sequence to be the label of this leaf's neighbor. For instance, the Prüfer code of the tree in Figure 1.1 is $(2,4,7,0,2,3)$.

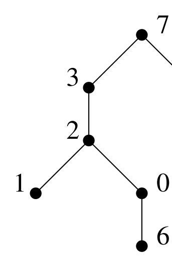

Figure 1.1
A labeled tree with Prüfer code $(2,4,7,0,2,3)$.

A bijection between $\mathscr{PK}_n$ and $\mathscr{T}_{n+1}$ was constructed by Pollak [71] using Prüfer code.

Theorem 1.1 *For each parking function* $\mathbf{a} = (a_1, a_2, \ldots, a_n)$*, define the difference sequence* $(c_1, c_2, \ldots, c_{n-1})$ *by letting*

$$c_i = a_{i+1} - a_i \mod n+1. \tag{1.1}$$

Let $T(\mathbf{a})$ *be the labeled tree in* $\mathscr{T}_{n+1}$ *whose Prüfer code is* $(c_1, c_2, \ldots, c_{n-1})$*. Then the map* $\mathbf{a} \to T(\mathbf{a})$ *is a bijection from* $\mathscr{PK}_n$ *to* $([n]_0)^{n-1}$.

Sketch of Proof. The inverse equations of (1.1) can be written as

$$a_i = a_1 + c_1 + \cdots + c_{i-1}, \quad \mod n+1 \qquad 2 \le i \le n. \tag{1.2}$$

That is, a parking function is determined by its difference sequence if a_1 is known. The following algorithm describes how to find an a_1 for each $(c_1, c_2, \ldots, c_{n-1}) \in \{0, 1, \ldots, n\}^{n-1}$ such that $(a_1, a_2, \ldots, a_n)$ determined by equations (1.2) is a parking function.

Algorithm to determine a_1. Given $(c_1, c_2, \ldots, c_{n-1}) \in \{0, 1, \ldots, n\}^{n-1}$,

1. Let
$$\begin{aligned} h_1 &= 0, \\ h_i &= c_1 + \cdots + c_{i-1} \mod n+1, \qquad 2 \le i \le n. \end{aligned}$$
2. Let $\mathbf{r}(h) = (r_0, \ldots, r_n)$ be the *specification* of h, i.e., $r_i = \text{card}\{j : h_j = i\}$. Let $R_j(h) = r_0 + \cdots r_j - j - 1$ for $0 \le j \le n$.
3. Find d to be the smallest index such that $R_d(h) = \min\{R_j(h) : 0 \le j \le n\}$. Then
$$a_1 = n - d.$$

The uniqueness of a_1 follows from the fact that the number of parking functions of length n is equal to the number of Prüfer codes of length $n-1$. □

Example 1.2 *For the Prüfer code* $(2,4,7,0,2,5)$*, we have* $h = (0,2,6,5,5,7,4)$ *and hence* $\mathbf{r}(h) = (1,0,1,0,1,2,1,1)$.

The sequence $R(h)$ *is given by* $(0,-1,-1,-2,-2,-1,-1,-1)$*, where the minimal value of* $R_j(h)$ *is reached at* $R_3(h) = R_4(h) = -2$*. The smallest index* d *is 3. Hence we have* $a_1 = n-3 = 4$ *which recovers the parking function* $(4,6,2,1,1,3,0)$.

A zero in the Prüfer code indicates a pair of consecutive like numbers in the parking function. Consequently Theorem 1.1 leads to an explicit formula for the enumerator of parking functions by the number of pairs of like consecutive numbers. Precisely, we have

Corollary 1.3 *The equality*
$$\sum_{\alpha \in \mathscr{PK}_n} q^{|\{i \,:\, a_i = a_{i+1}\}|} = (q+n)^{n-1}.$$
holds, where $|S|$ *is the cardinality of a set* S.

1.2.2 Inversions of labeled trees

Let $\mathbf{a} = (a_1, \ldots, a_n)$ be a parking function of length n. Consider again the one-way street parking scenario, and assume that the car C_i parked at space p_i, for $1 \le i \le n$.

Then $p_1 p_2 \cdots p_n$ is a permutation on the letters $\{0,1,\ldots,n-1\}$. Let $D(\mathbf{a})$ be the total *displacement*, i.e.

$$D(\mathbf{a}) = \sum_{i=1}^{n}(p_i - a_i) = \binom{n}{2} - \sum_{i=1}^{n} a_i. \tag{1.3}$$

Then $D(\mathbf{a})$ is the total number of failed trials before all the cars find their parking spaces. In the language of hashing in computer algorithms, $D(\mathbf{a})$ represents the number of linear probing. See Section 1.3.1 for more discussion.

Define the *displacement-enumerator of parking functions* as the polynomial

$$P_n(q) = \sum_{\mathbf{a}\in\mathscr{P}\mathscr{K}_n} q^{D(\mathbf{a})} = q^{\binom{n}{2}} \sum_{\mathbf{a}\in\mathscr{P}\mathscr{K}_n} q^{-(a_1+\cdots+a_n)}. \tag{1.4}$$

The degree of $P_n(q)$ is $\binom{n}{2}$. This polynomial also enumerates labeled trees by some important tree statistics, one of which is the number of *inversions*.

Let $T \in \mathscr{T}_{n+1}$. View vertex 0 as the root of T. If $\{i, j\}$ is an edge of T and j lies on the unique path connecting 0 to i, we say that i is the *predecessor* of j and j is a *successor* of i. In a rooted tree the degree of a vertex i is the number of its successors. A leaf is a non-root vertex with no successor.

An *inversion* of the tree T is a pair (i, j) for which $i < j$ and j lies on the unique path connecting 0 to i. Let $\mathrm{inv}(T)$ denote the number of inversions of T. The *inversion enumerator $I_n(q)$ of labeled trees on $n+1$ vertices* is defined as the polynomial

$$I_n(q) = \sum_{T\in\mathscr{T}_{n+1}} q^{\mathrm{inv}(T)},$$

which is also called the inversion enumerator of labeled forests on $[n]$.

Kreweras [56] studied polynomial systems satisfying a recurrence relation, and found several combinatorial interpretations, among which are the polynomials $I_n(q)$ and $P_n(q)$. In Kreweras paper, parking functions are called *suites majeures*.

Theorem 1.4 *1. The inversion enumerator $I_n(q)$ of labeled trees on $n+1$ vertices satisfies the recurrence relation*

$$\begin{aligned} I_1(q) &= 1, \\ I_{n+1}(q) &= \sum_{i=0}^{n} \binom{n}{i} (q^i + q^{i-1} + \cdots + 1) I_i(q) I_{n-i}(q). \end{aligned}$$

2. The displacement enumerator of parking functions satisfies the recurrence relation

$$\begin{aligned} P_1(q) &= 1, \\ P_{n+1}(q) &= \sum_{i=0}^{n} \binom{n}{i} (q^i + q^{i-1} + \cdots + 1) P_i(q) P_{n-i}(q). \end{aligned}$$

3. Consequently,

$$I_n(q) = P_n(q).$$

In fact, these polynomials are immediately related to connected graphs.

Theorem 1.5 *Let $C_n(q)$ be the edge-enumerator of connected graphs. Precisely,*

$$C_n(q) = \sum_G q^{|E(G)|-n},$$

where G ranges over all connected graphs (without loops or multiple edges) on $n+1$ labeled vertices, and $E(G)$ is the set of edges of G. Then

$$I_n(1+q) = P_n(1+q) = C_n(q).$$

Theorem 1.5 implies that the sum of $\binom{D(\mathbf{a})}{k}$ taken over all parking functions of length n, as well as the sum of $\binom{\mathrm{inv}(T)}{k}$ taken over all labeled trees on $n+1$ vertices, are equal to the total number of connected graphs with $n+k$ edges on $n+1$ labeled vertices. Analysis of these generating functions is essential in characterizing the evolution of random graphs, see Janson, Knuth, Łuczak and Pittle [48].

Using the exponential formula on graphs and connected graphs, one derives the following generating function identities, ([81]).

Theorem 1.6

1.

$$\sum_{n\geq 1} I_n(q)(q-1)^{n-1}\frac{x^n}{n!} = \sum_{n\geq 1} P_n(q)(q-1)^{n-1}\frac{x^n}{n!} = \log \sum_{n\geq 0} q^{\binom{n}{2}}\frac{x^n}{n!}. \qquad (1.5)$$

2.

$$\sum_{n\geq 0} I_n(q)(q-1)^{n-1}\frac{x^n}{n!} = \sum_{n\geq 0} P_n(q)(q-1)^{n-1}\frac{x^n}{n!} = \frac{\sum_{n\geq 0} q^{\binom{n+1}{2}}\frac{x^n}{n!}}{\sum_{n\geq 0} q^{\binom{n}{2}}\frac{x^n}{n!}}. \qquad (1.6)$$

Based on Theorem 1.4, Kreweras constructed recursively a bijection that carries the total displacement $D(\mathbf{a})$ of parking functions to the number of inversions of labeled trees. A more direct and elegant construction was given by Knuth [51], which we describe next.

<u>A Bijection $\phi : PK_n \to \mathscr{T}_{n+1}$ such that $D(\mathbf{a}) = \mathrm{inv}(\phi(\mathbf{a}))$.</u>

Let $\mathbf{a} = (a_1,\ldots,a_n) \in \mathscr{PK}_n$ and p_i be the space that car C_i occupies. Let $p_i' = p_i + 1$, hence $\mathbf{p}' = p_1' p_2' \cdots p_n'$ is a permutation of length n. Let $\mathbf{q} = q_1 q_2 \cdots q_n$ be the inverse permutation of $\mathbf{p}'$, that is, car C_{q_i} is parked at the space $i-1$.

1. Construct an auxiliary tree by letting the predecessor of vertex k be the first element on the right of k and larger than k in the permutation $q_1 q_2 \cdots q_n$; if there is no such element, let the predecessor be 0.

2. Make a copy of the auxiliary tree. Then relabel the nonzero vertices of the new tree by proceeding as follows, in preorder (i.e., any vertex is processed before its successors): if the label of the current vertex was k in the auxiliary tree, swap its current label with the label that is currently $(p'_k - a_k)$-th smallest in the subtree rooted at the current vertex. The final tree is $\phi(\mathbf{a})$.

Example 1.7 *Let* $\mathbf{a} = (4,0,1,0,3,6,4)$. *Then* $\mathbf{p} = p_1 \cdots p_n = 4\ 0\ 1\ 2\ 3\ 6\ 5$ *and* $\mathbf{p}' = p'_1 p'_2 \cdots p'_n = 5\ 1\ 2\ 3\ 4\ 7\ 6$. *Consequently, the inverse of the* $\mathbf{p}'$ *is* $\mathbf{q} = q_1 \cdots q_n = 2\ 3\ 4\ 5\ 1\ 7\ 6$. *Figure 1.2 shows the auxiliary tree and the final tree defined by Knuth's bijection. One checks that* $D(\mathbf{a}) = \mathrm{inv}(\phi(\mathbf{a})) = 3$.

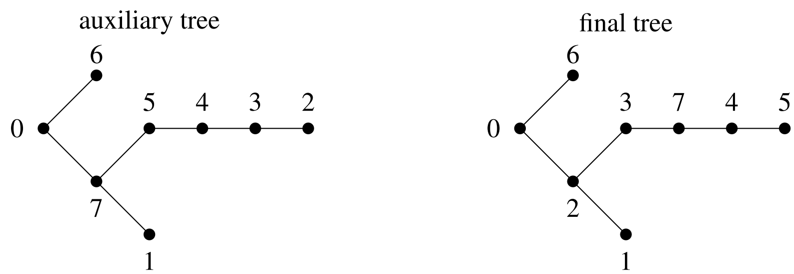

Figure 1.2
The auxiliary tree and the tree $\phi(\mathbf{a})$ corresponding to $\mathbf{a} = (4,0,1,0,3,6,4)$.

To reverse the procedure, we can reconstruct the auxiliary tree from a labeled tree $T \in \mathscr{T}_{n+1}$ by proceeding in preorder to swap the label of each vertex with the largest label currently in its subtree. Comparing the auxiliary tree and the final tree we could obtain the values of $p'_k - a_k$ for each $1 \le k \le n$. The permutation $q_1 \cdots q_n$ can be read from the auxiliary tree in postorder, which is recursively defined by

- If a tree T is null, then the empty list is the postorder listing of T.
- If T comprises a single node, that node itself is the postorder list of T.
- Otherwise, T has root r and the successors of r are $t_1 > t_2 > \cdots > t_k$. Let T_i be the subtree with root t_i. The postorder listing of T is the nodes of T_1 in postorder, ..., the nodes of T_k in postorder, all followed by the root r.

For example, the postorder of the auxiliary tree in Figure 1.2 is $2\ 3\ 4\ 5\ 1\ 7\ 6\ (0)$, which is $\mathbf{q} = q_1 \cdots q_n$. Taking the inverse of $\mathbf{q}$, one recovers $p'_1 p'_2 \cdots p'_n$, and hence $a_1 \cdots a_n$.

Knuth's bijection has another nice property: in the tree $\phi(\mathbf{a})$, if a vertex with label t is labeled k in the auxiliary tree, then there are $p_k - a_k$ nodes with smaller labels than t in the subtree of $\phi(\mathbf{a})$ rooted at t. In a labeled tree T, call a vertex v *leader* if it is the smallest among all the vertices of the subtree of T rooted at b. In particular, every leaf is a leader. Denote by $\text{lead}(T)$ the number of leaders in a labeled tree T. On the other hand, for a parking function $\mathbf{a}$ we say a car C_i is *lucky* if $p_i = a_i$, that is, C_i is parked at its preferred space. Let $\text{lucky}(\mathbf{a})$ denote the number of lucky cars. Then Knuth's bijection ϕ satisfies $\text{lucky}(\mathbf{a}) = \text{lead}(\phi(\mathbf{a}))$.

The statistics "lucky($\mathbf{a}$)" for parking functions and "lead(T)" for labeled trees were studied by Gessel and Seo [37], and Seo and Shin [73], respectively, where they gave an explicit formula for the corresponding enumerators as

$$\sum_{\mathbf{a}\in\mathscr{P}\mathscr{K}_n} u^{\text{lucky}(\mathbf{a})} = \sum_{T\in\mathscr{T}_{n+1}} u^{\text{lead}(T)} = u\prod_{i=1}^{n-1}\big(i+(n-i+1)u\big).$$

A combinatorial explanation of the relation between the inversion-enumerator of trees and the edge-enumerator of connected graphs was given by Gessel and Wang [38], who used depth-first search (DFS) to partitions the set of connected graphs on $n+1$ labeled vertices into disjoint Boolean blocks, each of which is represented by a labeled tree. This idea is crucial for establishing the connection between G-parking functions and the Tutte polynomial of G, (cf. Section 1.5.)

Let G be a connected graph on $[n]_0$. The DFS algorithm is applied to G and returns a certain spanning tree $T = \text{DFS}(G)$ by the following procedure. We start at vertex 0 (which is viewed at the root of the tree), and at each step we go to the greatest adjacent unvisited vertex if there is one, otherwise, we backtrack. For example, from the graph on the left of Figure. 1.3, we get the spanning tree on the right.

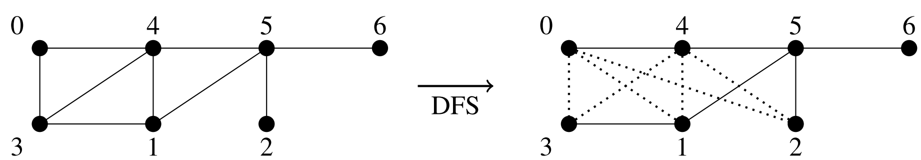

Figure 1.3
The spanning tree found by DFS on a connected graph.

Fix a labeled tree $T \in \mathscr{T}_{n+1}$. Let $\mathscr{G}(T)$ be the set of connected graphs G for which $\text{DFS}(G) = T$. Define a set $\mathscr{E}(T)$ of edges not in T whose elements are in one-to-one correspondence with the inversions of T. More precisely, with every inversion (j,k), $(j > k > 0)$, associate the edge between k and the predecessor of j. For the tree on the right of Figure 1.3, the edges of $\mathscr{E}(T)$ are indicated by dotted lines.

Gessel and Wang characterized the set of connected graphs in $\mathscr{G}(T)$.

Theorem 1.8 *$\mathscr{G}(T)$ consists of those graphs obtained from T by adjoining an arbitrary subset of edges in $\mathscr{E}(T)$.*

An immediate corollary of Theorem 1.8 is the equation

$$\sum_{G \in \mathscr{G}(T)} q^{|E(G|)-n} = (1+q)^{\mathrm{inv}(T)}.$$

Summing over all trees $T \in \mathscr{T}_{n+1}$, we have

$$C_n(q) = \sum_{G \text{ connected}} q^{|E(G)|-n} = \sum_{T \in \mathscr{T}_{n+1}} q^{\mathrm{inv}(T)} = I_n(1+q).$$

In the next subsection we will see that applying the breadth-first search on the connected graphs leads to the equation between $P_n(1+q)$ and $C_n(q)$.

1.2.3 Graph searching algorithms

For a sequence $\mathbf{a} = (a_1, \ldots, a_n)$ with $0 \le a_i < n$, its *specification* is the vector $\vec{r}(\mathbf{a}) = (r_0, \ldots, r_{n-1})$ where $r_i = |\{j \ : \ a_j = i\}|$. If $\mathbf{a}$ is a parking function, then the vector $\vec{r}(\mathbf{a})$ is *balanced*, that is,

$$r_0 + \cdots + r_i - i - 1 \quad \ge \quad 0 \qquad \text{for } 0 \le i < n, \tag{1.7}$$

$$r_0 + \cdots + r_{n-1} \quad = \quad n. \tag{1.8}$$

In addition, define the *order permutation* $\sigma(\mathbf{a}) = \sigma_1 \sigma_2 \cdots \sigma_n$ of a parking function $\mathbf{a}$ by letting

$$\sigma_i = |\{j \ : \ a_j < a_i, \text{ or } a_j = a_i \text{ and } j \le i\}|. \tag{1.9}$$

In other words, σ_i is the position of the entry a_i in the non-decreasing rearrangement of $\mathbf{a}$. For example, for $\mathbf{a} = (2\ 0\ 3\ 0\ 4\ 8\ 1\ 5\ 4)$, the specification is $\vec{r}(\mathbf{a}) = (2,1,1,1,4,1,0,0,1)$, and the order permutation is $\sigma(\alpha) = 4\ 1\ 5\ 2\ 6\ 9\ 3\ 8\ 7$. Clearly a parking function determines its specification and order permutation. Conversely, knowing the vector $\vec{r}(\mathbf{a})$ and the permutation $\sigma(\mathbf{a})$, we can easily recover $\mathbf{a}$ by replacing i in $\sigma(\mathbf{a})$ by the i-th smallest term in the list $0^{r_0} 1^{r_1} \cdots (n-1)^{r_{n-1}}$.

Not every pair of a vector $\vec{r}$ and a permutation σ can be the specification and the order permutation of a parking function. The vector and the permutation must be *compatible* with each other, that is, the terms in the inverse σ^{-1} of σ are increasing on every interval with the indices $\{1 + \sum_{i=0}^{k-1} r_i, 2 + \sum_{i=0}^{k-1} r_i, \ldots, \sum_{i=0}^{k} r_i\}$ (if $r_k \neq 0$). Equivalently, the terms $1 + \sum_{i=0}^{k-1} r_i, 2 + \sum_{i=0}^{k-1} r_i, \ldots, \sum_{i=0}^{k} r_i$ appear from left to right in σ.

Let $\mathscr{C}_n$ be the set of all pairs $(\vec{r}, \sigma)$ with $\vec{r} \in \mathbb{N}^n$ and σ a permutation of length n compatible with $\vec{r}$.

Theorem 1.9 *The map $\rho : \mathbf{a} \to (\vec{r}(\mathbf{a}), \sigma(\mathbf{a}))$ is a bijection from $\mathscr{P}\mathscr{K}_n$ to $\mathscr{C}_n$.*

On the other hand, there are many ways to construct bijections between $\mathscr{C}_n$ and $\mathscr{T}_{n+1}$, the set of labeled trees with the vertex set $[n]_0$. Each such bijection, combined with Theorem 1.9, gives a bijection between $\mathscr{P}\mathscr{K}_n$ and $\mathscr{T}_{n+1}$.

Theorem 1.10 *The set $\mathscr{C}_n$ is in one-to-one correspondence with $\mathscr{T}_{n+1}$.*

Here we introduce a family of bijections between $\mathscr{C}_n$ and $\mathscr{T}_{n+1}$, each of which is determined by a choice function and corresponds to a searching algorithm on trees. Generally speaking, a searching algorithm checks the vertices of a tree one-by-one, starting with the root 0. At each step, we pick a new vertex that is connected to the "checked" vertices. The choice function would tell us which new vertex to pick.

Let Π be the set of all ordered pairs (F,W) such that F is a tree whose vertex set $V(F)$ is a subset of $[n]_0$ containing the root 0, and $\emptyset \neq W \subseteq \text{Leaf}(F)$ where $\text{Leaf}(F)$ is the set of leaves of F. A *choice function* γ is a function from Π to $[n]$ such that $\gamma(F,W) \in W$. Some choice functions are described in Examples 1.12–1.15.

Fix a choice function γ. Given a tree $T \in \mathscr{T}_{n+1}$, we define a linear order of the vertices $V(\gamma) = v_0, v_1, v_2, \ldots, v_n$. First, set $v_0 = 0$. For each $0 < i < n$, assuming $v_0, \ldots, v_{i-1}$ are determined. Let $W_i = \{v$: the predecessor of v is in $\{v_0, \ldots, v_{i-1}\}$ $\}$, and F_i be the subtree obtained from T by restricting to $W_i \cup \{v_0, \ldots, v_{i-1}\}$. Then let $v_i = \gamma(F_i, W_i)$.

For each v_i, order the successors of v_i from small to large. Let π_γ be obtained by reading the successors of v_0, followed by successors of v_1, then successors of v_2, and so on. Finally, let $\vec{r}_\gamma = (r_0, r_1, \ldots, r_{n-1})$ where r_i is the number of successors of the vertex v_i.

Theorem 1.11 *The map $\phi_\gamma : T \to (\vec{r}_\gamma, \pi_\gamma^{-1})$ described above is a bijection from $\mathscr{T}_{n+1}$ onto $\mathscr{C}_n$, where π_γ^{-1} is the inverse of π_γ.*

We explain the bijections with some explicit examples of choice functions, and apply each bijection to the tree T in Figure 1.4.

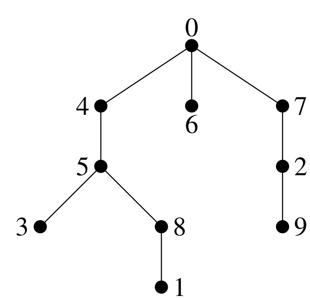

Figure 1.4
A tree in $\mathscr{T}_{10}$.

Example 1.12 The vertex-adding order.

Let $\gamma(F,W)$ be the vertex in W with minimal label. Then the vertex ordering is $V(\gamma) =$ 0 4 5 3 6 7 2 8 1 9. Hence

$$\vec{r}_\gamma = (3,1,2,0,0,1,1,1,0), \quad \pi_\gamma = 467538291.$$

Example 1.13 Depth-first search order.
Also known as the preorder traversal, the depth-first order is the list of vertices in the order that they were first visited by the DFS. Here we adapt the version of DFS described in the end of Section 1.2.2.

The choice function γ_{df} is defined as $\gamma_{df}(F,W) = v$ where v is the minimal element of W under the depth-first search order. Then the vertex ordering defined by γ_{df} is $V(df) =$ 0 7 2 9 6 4 5 8 1 3. Hence

$$\vec{r}_{\gamma_{df}} = (3,1,1,0,0,1,2,1,0), \quad \pi_{\gamma_{df}} = 467295381.$$

Example 1.14 Breadth-first search order.
Breadth-first search (BFS) is another commonly used graph searching algorithm, which begins at the root and explores all the neighboring nodes before going to the next node. The order that the vertices are visited under the BSF is called the BFS order, and denoted by $<_{bf}$. In a labeled tree $T \in \mathscr{T}_{n+1}$, let $\text{level}(i)$ be the distance of a node i to the root 0. A simple version of the BFS order of the vertices of T is to let $i <_{bf} j$ if $\text{level}(i) < \text{level}(j)$, or if $\text{level}(i) = \text{level}(j)$ but $i < j$.

The choice function γ_{bf} is defined as $\gamma_{bf}(F,W) = v$ where v is the minimal element of W under the breadth-first search order $<_{bf}$. Then for the tree in Figure 1.4, the vertex ordering defined by γ_{bf} is $V(\gamma_{df}) =$ 0 4 6 7 2 5 3 8 9 1. Hence

$$\vec{r}_{\gamma_{bf}} = (3,1,0,1,1,2,0,1,0), \quad \pi_{\gamma_{bf}} = 467529381.$$

Example 1.15 Breadth-first search order with a queue.
This is a variation of the BFS order in Example 1.14. It corresponds to an implementation of BFS with a queue structure. It is a vertex ordering, denoted by $<_{bfq}$, which can be defined explicitly by letting $i <_{bfq} j$ if (1) $\text{level}(i) < \text{level}(j)$, or (2) $\text{level}(i) = \text{level}(j)$ and $\text{pre}(i) <_{bfq} \text{pre}(j)$, where $\text{pre}(i)$ is the predecessor of i; or (3) $\text{pre}(i) = \text{pre}(j)$ and $i < j$.

The choice function $\gamma_{bfq}(F,W)$ always chooses the minimal element of W under the order $<_{bfq}$. Under this choice function the vertex ordering is 0 4 6 7 5 2 3 8 9 1. *And*

$$\vec{r}_{\gamma bf,q} = (3,1,0,1,2,1,0,1,0), \qquad \pi_{\gamma_{bfq}} = 467523891.$$

Note that in this case, the permutation $\pi_{\gamma_{bfq}}$ is the same as the vertex ordering, (after removing 0.)

Theorem 1.9 and the bijections given in Theorem 1.11 have some interesting implications. For example, combining the map ρ of Theorem 1.9 and any bijection of Theorem 1.11, we get that the number of parking functions with k entries equal to 0 is equal to the number of labeled trees on $n+1$ vertices whose root vertex has degree k, which in turn equals the number of rooted forests on $[n]$ with k tree components. The latter is well-known to be $\binom{n-1}{k-1}n^{n-k}$. Written in terms of generating functions, we have

Corollary 1.16 *The enumerator of parking functions by the number of elements equal to* 0 *is* $x(x+n)^{n-1}$*, i.e.,*

$$\sum_{\mathbf{a}\in\mathscr{P}\mathscr{K}_n} x^{|\{i\ a_i=0\}|} = x(x+n)^{n-1}.$$

Next we describe a special implementation of the Breadth-first search algorithm with a queue structure, which finds a particular spanning tree for any connected graph. The algorithm, combining with Theorems 1.9 and 1.11, gives the combinatorial explanation of the equation $P_n(1+q) = C_n(q)$.

Let G be a connected graph on $[n]_0$. The BFS algorithm is described as a queue Q that starts at vertex 0. At each stage we take the vertex x at the head of Q, remove x from Q, and add all unvisited neighbors y of x to Q in numerical order. We will call that operation "processing x". We continue the above procedure until the queue is empty. The spanning tree $T = BFS(G)$ is obtained by adding all edges of the form $\{x,y\}$, where x is the vertex being processed, and y is an unvisited neighbor of x. For the connected graph G in Figure 1.3, the BFS gives the spanning tree shown on the right-hand side of Figure 1.5.

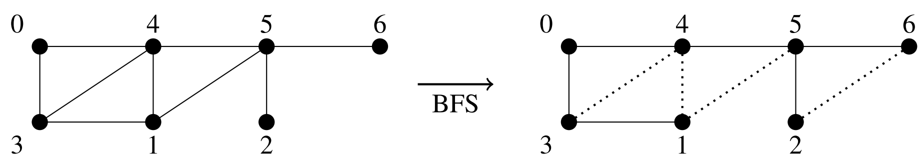

Figure 1.5
Spanning tree of Figure 1.3 found by BFS, where the dotted edges belong to $\mathscr{E}_1(T)$.

Let r_t be the number of vertices found by the t-th vertex processed, for $t = 0,\ldots,n$, which in our example, are $(r_0,r_1,\ldots,r_6) = (2,1,1,0,2,0,0)$. Note that r_n is always 0. Let q_t be the size of the queue after the t^{th} vertex is processed. Then $q_0 = 1$ and $q_t = q_{t-1} + r_{t-1} - 1$, which, in our example, are 1, 2, 2, 2, 1, 2, 1, 0.

For the example in Figure 1.5, the queue Q, r_t and q_t at each stage t are listed in the following table.

t	0	1	2	3	4	5	6	7
Q	0	3 4	4 1	1 5	5	2 6	6	$\emptyset$
r_t	2	1	1	0	2	0	0	
q_t	1	2	2	2	1	2	1	0

A necessary and sufficient condition for the graph being connected is that the sequence $\{q_t\}$ satisfies

$$q_{n+1} = 0, \qquad \text{and} \qquad q_i > 0 \ \text{ for } i \leq n. \tag{1.10}$$

In terms of r_t, the above constraints are equivalent to

$$r_0 + \cdots r_{i-1} \geq i \text{ for } i < n, \qquad r_0 + \cdots + r_{n-1} = n.$$

In other words, $\vec{r} = (r_0, \ldots, r_{n-1})$ is balanced. Also note that the queue uniquely determines a permutation π of length n, namely, the order that the vertices are processed in the queue. One can read it from the head of the queue. In our example, it is 341526. The vector $\vec{r}$ and the permutation π^{-1} are compatible. Actually, they are uniquely determined by the spanning tree T found by the BFS.

Let $\mathbf{a}(T)$ be the parking function that corresponds to the pair $(\vec{r}, \pi^{-1})$ by the bijection ρ of Theorem 1.9, where $T = BFS(G)$. One easily computes that the displacement of $a(T)$ is $D(\mathbf{a}(T)) = \binom{n}{2} - \sum_{i=0}^{n-1} ir_i$. In our example, $\mathbf{a}(G) = (1,4,0,0,2,4)$, and $D(\mathbf{a}(G)) = 4$.

Let $\mathscr{G}_1(T)$ be the set of connected graphs for which the spanning tree found by the BFS is T. The following crucial observation is due to Spencer [76]: An edge $\{i, j\}$ can be added to T without changing the spanning tree under the BFS if and only if in the queue, when the first of the two vertices is processed, the other is currently in the queue. Let $\mathscr{E}_1(T)$ be the set of all such edges. In our example, $\mathscr{E}_1(T) = \{\{3,4\}, \{4,1\}, \{1,5\}, \{2,6\}\}$. See Figure 1.5 to compare with the table of the queue.

It follows that

Theorem 1.17 *$\mathscr{G}_1(T)$ consists of these graphs obtained from T by adjoining an arbitrary subset of edges in $\mathscr{E}_1(T)$.*

Thus

$$\sum_{G \in \mathscr{G}_1(T)} q^{|E(G)|-n} = (1+q)^{|\mathscr{E}_1(T)|}.$$

Now we compute $|\mathscr{E}_1(T)|$. From the queue Q, we have

$$|\mathscr{E}_1(T)| = \sum_{i=1}^{n} (q_i - 1) = \sum_{i=1}^{n} (r_0 + \cdots + r_{i-1} - i) = \sum_{i=0}^{n-1} (n-i) r_i - \binom{n+1}{2}.$$

Since $\sum_{i=0}^{n-1} r_i = n$, the number of the last formula equals

$$n \sum_{i=0}^{n-1} r_i - \sum_{i=0}^{n-1} ir_i - \binom{n+1}{2} = \binom{n}{2} - \sum_{i=0}^{n-1} ir_i = D(\mathbf{a}(T)).$$

Thus

$$\sum_{G \in \mathscr{G}_1(T)} q^{|E(G)|-n} = (1+q)^{D(\mathbf{a}(T))}.$$

Summing over all trees T on $[n]_0$, we obtain

$$C_n(q) = \sum_{G \text{ connected}} q^{|E(G)|-n} = \sum_{T \in \mathscr{T}_{n+1}} (1+q)^{D(\mathbf{a}(T))} = P_n(1+q).$$

1.2.4 External activity of labeled trees

Another combinatorial statistic of labeled trees that corresponds to the displacement of parking functions is the external activity, a notion proposed by Tutte in defining a bivariate polynomial, called *Tutte polynomial*, for undirected graphs. Tutte polynomial plays an important role in graph enumeration. The evaluations of Tutte polynomial at various points give the numbers of spanning trees, spanning forests, connected subgraphs, acyclic orientations, etc. Here we just recall the definition of external activity. In Section 1.5.3 we will present a much stronger relation between Tutte polynomial and general parking functions.

Consider the complete graph K on $[n]_0$. Fix a total ordering of its edges. For a tree $T \in \mathscr{T}_{n+1}$, an edge $e \in K - T$ is *externally active* if it is the smallest edge in the unique cycle contained in $T \cup \{e\}$. The *external activity* $ea(T)$ is the number of externally active edges of T. Define the *external activity enumerator* of trees in $\mathscr{T}_{n+1}$ as

$$EA_n(q) = \sum_{T \in \mathscr{T}_{n+1}} q^{ea(T)}.$$

Then

$$EA_n(q) = I_n(q) = P_n(q).$$

The equation $EA_n(q) = I_n(q)$ was proved by Björner [13] using his results on shellability and homology in matroids. Beissinger [9] constructed a bijection from parking functions to labeled trees that carries the displacement of parking functions to the external activity of trees.

1.2.5 Sparse connected graphs

Theorem 1.5 has a nice application in graphical enumeration. Let $c(n+1,k)$ be the number of labeled connected graphs on $n+1$ vertices with exactly $n+k$ edges. For example, $c(n+1,0) = (n+1)^{n-1}$ by Cayley's formula. Theorem 1.5 implies that

$$c(n+1,k) = \sum_j p_j \binom{j}{k},$$

where p_j is the number of parking functions that have displacement $D(\mathbf{a}) = j$. Let $F_k(n)$ be the kth falling factorial moment of $D(\mathbf{a})$, i.e.,

$$F_k(n) = \frac{1}{(n+1)^{n-1}} \sum_{\mathbf{a} \in \mathscr{PK}_n} (D(\mathbf{a}))_k = \frac{1}{(n+1)^{n-1}} \sum_{\mathbf{a} \in \mathscr{PK}_n} k! \binom{D(\mathbf{a})}{k},$$

where $(n)_k$ is the falling factorial $n(n-1)\cdots(n-k+1)$. It follows that

Theorem 1.18

$$c(n+1,k) = \frac{(n+1)^{n-1}}{k!} F_k(n).$$

Theorem 1.18 can also be obtained by using a result of Spencer [76] that

$$\frac{c(n+1,k)}{c(n+1,0)} = E\left[\binom{M}{k}\right]$$

where M is a certain random variable defined on all labeled trees with $n+1$ vertices with uniform distribution. This formula, together with the bijection between labeled trees and parking functions, yields Theorem 1.18.

For $k = 1,2$, the formulas for $F_k(n)$ are computed in [36, 52] as

$$F_1(n) = \frac{1}{2}\sum_{i=2}^{n}\binom{n}{i}\frac{i!}{(1+n)^{i-1}},$$

$$F_2(n) = \frac{n(n-1)(n-2)}{24(n+1)^2}\big(15Q_3(n+1,n-3)+7Q_2(n+1,n-3)+2Q_1(n+1,n-3)\big),$$

where

$$Q_r(m,n) = \sum_{k\geq 0}\binom{r+k}{k}\frac{n(n-1)\cdots(n-k+1)}{m^k}. \tag{1.11}$$

Explicit formulas for higher moments are computed by Kung and Yan [58, Theorem 7.1] by solving a recursion based on a combinatorial decomposition of parking functions. The decomposition is the key idea in connecting parking functions to Gončarov polynomials, a system of polynomials from interpolation theory that provides a natural tool for studying the algebraic properties of parking functions.

In [84] Wright found the following asymptotic formula: for fixed k and n tending to infinity,

$$c(n+1,k) = \rho_{k-1}(n+1)^{n-1+3k/2}\big(1+O(n^{-1/2})\big).$$

The Wright constants ρ_k is given by

$$\rho_k = \frac{\pi^{1/2}2^{(1-3k)/2}\sigma_k}{\Gamma((3k/2)+1)},$$

where σ_k is defined by a second order recursion

$$\sigma_{k+1} = \frac{3(k+1)\sigma_k}{2} + \sum_{s=1}^{k-1}\sigma_s\sigma_{k-s} \qquad (k\geq 2),$$

with initial values $\sigma_0 = 1/4$, $\sigma_1 = 5/16$, and $\sigma_2 = 15/16$. The first several values of ρ_k are $\rho_0 = \sqrt{2\pi}/4$, $\rho_1 = 5/24$, and $\rho_2 = 5\sqrt{2\pi}/2^8$. Using combinatorial analysis, Kung and Yan [58] gave another formula for the Wright constants ρ_k, which only involves a linear recurrence.

REMARK. Theorems 1.9 and 1.10 are due to Foata and Riordan [30], where they used the breadth-first search order of Example 1.15 to prove Theorem 1.10. A simpler version of choice functions was first appeared in Françon [31] as “selection procedures” that lead to a family of bijections between parking functions and labeled trees, each corresponding to a searching algorithm. The idea was extended to parking functions associated with graphs by Chebikin and Pylyavskyy [21], and Kostic and Yan [54]. Spencer [76] used the implementation of BFS with a queue to develop an exact formula for the number of labeled connected graphs on $[n]$ with $n-1+k$ edges (k fixed) in terms of appropriate expectations. Moving to asymptotics, Spencer showed that the expectations can be expressed in terms of a certain restricted Brownian motion. Spencer's approach was generalized by Kostic and Yan [54] to establish the relation between parking functions associated to a graph G and the Tutte polynomial of G, (cf. Section 1.5). Results of [54] also reveal the connections between external activities of the spanning trees of a graph G and the parking functions associated to G.

1.3 Many Faces of Parking Functions

The parking function is an object lying in the center of combinatorics and appearing in many discrete and algebraic structures. In this section we describe some most important examples and their implications.

1.3.1 Hashing and linear probing

An efficient method for storing and retrieving data in computer programming is known as *hashing* or *scatter storage technique*. It has a hash function h which assigns a hash value $h(K)$ to each item K. However, two or more items may have the same hash value and hence cause a hash collision. *Linear probing and insertion* is a simple and basic algorithm for resolving hash collisions by sequentially searching the hash table for a free location. It inserts n items in $m > n$ cells by the following rules. Begin with all the cells $(0,1,\ldots,m-1)$ empty. Then, for $1 \leq k \leq n$, insert the k-th item into the first nonempty cell in the sequence $h_k, h_k+1, h_k+2, \ldots \pmod{m}$, where h_k is the hash value of the k-th item and is in the range $0 \leq h_k < m$. For a comprehensive description of hashing, as well as other storage and retrieval methods, see Section 6.4 of [51].

In the above description, the sequence $(h_1,\ldots,h_n)$ is called the *hash function*. It is *confined* if the linear probing with $h_1,\ldots,h_n$ will leave the cell $m-1$ empty. One notes immediately that when $m = n+1$, the confined hash functions are exactly parking functions of length n.

If the k-th item is inserted into position p_k, then the quantity $D(\mathbf{h}) = \sum_{k=1}^{n} [(p_k - h_k) \mod m]$ is the number of linear probes, or the total displacement of the items from their hash addresses, which gives the name *displacement* of the corresponding statistic of parking functions. Let $D_{m,n}(q) = \sum_{\mathbf{h}} q^{D(\mathbf{h})}$, where $\mathbf{h}$ ranges over all m^n possible hash functions $(h_1,\ldots,h_n)$ with $0 \leq h_k < m$, and let $F_{m,n}(q)$ be the same sum restricted to confined hash functions. Using Pollak's argument with m cells, one notes that given $(h_1,\dots,h_n)$, the m hash sequences $\{(h_i+j) \mod m, 0\le j<m\}$ all have the same total displacement, and exactly $(m-n)/m$ of them are confined. Therefore $F_{mn}(q)=\frac{m-n}{m}D_{mn}(q)$, and the probability generating function for $D(\mathbf{h})$ satisfies

$$\frac{D_{mn}(q)}{D_{mn}(1)}=\frac{F_{mn}(q)}{F_{mn}(1)}. \tag{1.12}$$

Equation (1.12) allows us to reduce the computation of the probability distribution of $D(\mathbf{h})$ over random hash functions to that of confined hash functions, which is easier since the linear probing does not "wrap around" when the hash function is confined. In particular, we have $F_{n+1,n}(q)=P_n(q)$, the displacement enumerator of parking functions of length n. In general, for $m=n+r$, a confined hash function $\mathbf{h}=h_1,\dots,h_n$ leaves the cells $n_1<n_2<\cdots<n_r$ empty if and only if $\mathbf{h}$ is obtained by merging r parking functions of lengths $n_1,n_2-n_1-1,\dots,n_r-n_{r-1}-1$. Let $s_1=n_1$ and $s_i=n_i-n_{i-1}-1$ for $2\le i\le r$. Then $\sum_{i=1}^r s_i=n$, and

$$F_{n+r,n}(q)=\sum_{s_1+s_2+\cdots+s_r=n}\binom{n}{s_1,s_2,\dots,s_r}P_{s_1}(q)P_{s_2}(q)\cdots P_{s_r}(q). \tag{1.13}$$

Let

$$F(q,z)=\sum_{n\ge 0}P_n(q)\frac{z^n}{n!}.$$

It follows from Equation (1.13) that

$$\sum_{n\ge 0}F_{n+r,n}(q)\frac{z^n}{n!}=F(q,z)^r.$$

In other words, the distribution of total displacement for linear probing with random hash functions is determined by the exponential generating function of $P_n(q)$ for parking functions. Some identities related to $F(q,z)$ and its variations are given in Theorem 1.6.

In analyzing the performance of hashing as a storage method, one is interested in the expected value of $D(\mathbf{h})$ over all hash functions $h:[n]\to[m]$, assuming all are equally likely. This value is $D'_{m,n}(1)/D_{m,n}(1)$, which equals $F'_{m,n}(1)/F_{m,n}$. Using combinatorial analysis and the Lagrange inversion formula, one gets an explicit formula for the expected value of $D(\mathbf{h})$, (see e.g. [36, 52]),

Theorem 1.19 *The expected value of linear probes $D(\mathbf{h})$ as h varies over all hash functions from $[n]$ to $[m]$ $(n<m)$ is*

$$\frac{1}{2}\sum_{i=2}^{n}\binom{n}{i}i!m^{1-i}=\frac{1}{2}\left[\frac{n(n-1)}{m}+\frac{n(n-1)(n-2)}{m^2}+\cdots\right]=\frac{n}{2}(Q_0(m,n-1)-1),$$

where $Q_r(m,n)$ is defined in (1.11).

Knuth also computed the second factorial moments of $D(\mathbf{h})$, and obtained the following expected value of $D(\mathbf{h})(D(\mathbf{h})-1)$ in Formula (5.5) of [52].

$$\begin{aligned} & Exp[D(\mathbf{h})(D(\mathbf{h})-1)] \\ = \; & \frac{n(n-1)(n-2)}{12m^2}[15Q_3(m,n-3)+(4+3m-3n)Q_2(m,n-3) \\ & +(5-3m+3n)Q_1(m,n-3)]. \end{aligned} \tag{1.14}$$

Moment analysis and characterizations of limit distributions for the linear probes are carried out by Flajolet, Poblete and Viola [29], and Janson [47].

1.3.2 Lattice of noncrossing partitions

A *set partition* of $[n]$ is a family of pairwise disjoint nonempty subsets $B_1,\dots,B_k$ whose union is $[n]$. A partition $\{B_1,\dots,B_k\}$ is *noncrossing* if there are no elements $a<b<c<d$ such that $a,c\in B_i$, $b,d\in B_j$ and $i\neq j$. The study of noncrossing partitions can be traced back to Becker [8] under the name "planar rhyme schemes". The systematic study of noncrossing partitions began with Kreweras [55] and Poupart [69], and became one of the central topics in contemporary combinatorics.

Let π and π' be two partitions of $[n]$. We say that π is a *refinement* of π' if every block of π is contained in a block of π'. Refinement induces a partial order on the set of partitions: $\pi<\pi'$ if and only if π is a refinement of π'. Under this order there is a unique maximal element, the partition with only one block $[n]$. In addition any two partitions π and π' have a greatest lower bound $\pi\wedge\pi'$ whose blocks are the non-empty intersections of one block of π and one block of π'. It is easy to check that if both π and π' are noncrossing, then so is $\pi\wedge\pi'$. Hence the set of noncrossing partitions form a lattice, which is denoted NC_n. The lattice of noncrossing partitions enjoys a number of remarkable properties, and plays a fundamental role in the combinatorics of Coxeter groups and free probability. See [1] and [66] for references. Next is a list of basic properties of lattice of non-crossing partitions.

1. It is well-known that the number of noncrossing partitions of $[n]$ is the n-th Catalan number $C_n=\frac{1}{n+1}\binom{2n}{n}$.

2. NC_n is a graded of rank $n-1$. The rank of a noncrossing partition π is $n-|\pi|$, where $|\pi|$ is the number of blocks in π.

3. The number of noncrossing partitions with exactly k blocks is given by the Narayana number $N(n,k)$, where
$$N(n,k)=\frac{1}{n}\binom{n}{k}\binom{n}{k-1}.$$

4. The lattice NC_n has the minimum $\hat{0}=\{1\}\{2\}\cdots\{n\}$ and the maximum $\hat{1}=[n]$.

5. NC_n is self-dual, i.e., there is an isomorphism $\xi : NC_n \to NC_n$ such that $\pi < \sigma$ if and only if $\xi(\sigma) < \xi(\pi)$. The Kreweras complement operation provides such an isomorphism, [55].

A surprising connection between parking functions and the lattice of noncrossing partitions was revealed by Stanley [80], who constructed an edge labeling of NC_{n+1} whose maximal chains are labeled by the set $\mathscr{PK}_n$, The labeling also leads to a local action of the symmetric group $\mathfrak{S}_n$ on NC_{n+1}.

In a locally finite poset P, an edge of P is a pair (u,v) such that v covers u, i.e., $u < v$ and there is no element t satisfying $u < t < v$. For NC_{n+1}, a partition π' covers partition π if and only if π' is obtained from π by merging two blocks into one. Let (π, π') be an edge of NC_{n+1} where two blocks B and B' of π are merged to form one block of π'. Suppose that $\min B < \min B'$. Define

$$\Lambda(\pi, \pi') = \max\{i \in B \;:\; i < \min B'\}.$$

For instance, if $B = \{2,3,6,16\}$ and $B' = \{8,9,10,14\}$, then $\Lambda(\pi,\pi') = 6$. This labeling is well-defined since the set $\{i \in B \;:\; i < \min B'\}$ is nonempty.

Figure 1.6 shows the labeling Λ on the lattice NC_3.

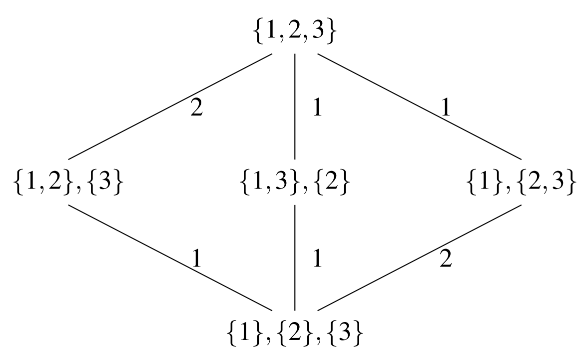

Figure 1.6
The labeling Λ on NC_3.

The labeling Λ of the edges of NC_{n+1} extends in a natural way to a labeling of the maximal chains. Namely, given a maximal chain $\mathfrak{m} : \hat{0} = \pi_0 < \pi_1 < \cdots < \pi_n = \hat{1}$ where π_i covers π_{i-1}, we set

$$\Lambda(\mathfrak{m}) = (\Lambda(\pi_0,\pi_1), \Lambda(\pi_1,\pi_2), \ldots, \Lambda(\pi_{n-1},\pi_n)).$$

Note that $\Lambda(\pi_i, \pi_{i+1}) \in [n]$. Adjust the values by letting

$$\Lambda_1(\mathfrak{m}) = (\Lambda(\pi_0,\pi_1) - 1, \Lambda(\pi_1,\pi_2) - 1, \ldots, \Lambda(\pi_{n-1},\pi_n) - 1).$$

Theorem 1.20 *The label* $\Lambda_1(\mathfrak{m})$ *of the maximal chains of* NC_{n+1} *consists of the parking functions of length n, each occurring once.*

As an immediate consequence, we recover a result of Kreweras [55].

Corollary 1.21 *The number of maximal chains of noncrossing partitions of* $\{1,2,\ldots,n+1\}$ *is* $(n+1)^{n-1}$.

The labeling Λ induces an R-labeling by letting $\Lambda^*(\pi,\sigma) = |\pi| - \Lambda(\pi,\sigma)$, as in the sense of [78, Def. 3.13.1]. More precisely, for every interval $[\pi,\pi']$ of NC_{n+1}, there is a unique maximal chain $\mathfrak{m} : \pi = \pi_0 < \pi_1 < \cdots < \pi_j = \pi'$ such that

$$\Lambda^*(\pi_0,\pi_1) \leq \Lambda^*(\pi_1,\pi_2) \leq \cdots \leq \Lambda^*(\pi_{j-1},\pi_j).$$

For any poset equipped with an R-labeling, there is a general theorem [78, Theorem 3.13.2] that allows us to enumerate the labeling of maximal chains with a given descent set in terms of the rank-selected Möbius invariants. Applying this theorem to NC_{n+1} and parking functions yields the following result.

For a finite graded poset P of rank n with $\hat{0}$ and $\hat{1}$ and with rank function ρ, let S be a subset of $[n-1]$ and denote $\alpha_P(S)$ to be the number of chains $\hat{0} = t_0 < t_1 < \cdots < t_s = \hat{1}$ of P such that $\{\rho(t_1),\rho(t_2),\ldots,\rho(t_{s-1})\} = S$. The function $\alpha_P(S)$ is called the *flag f-vector* of P. Further define

$$\beta_P(S) = \sum_{T\subseteq S} (-1)^{|S-T|}\alpha_P(T),$$

or equivalently,

$$\alpha_P(S) = \sum_{T\subseteq S} \beta_P(T).$$

The function β_P is called the *flag h-vector* of P.

For a parking function $\mathbf{a} = (a_1,\ldots,a_n)$, define the *descent set* $\mathrm{Des}(\mathbf{a})$ by letting

$$\mathrm{Des}(\mathbf{a}) = \{i \,:\, a_i > a_{i+1}\}.$$

Theorem 1.22 *Let* $S \subseteq [n-1]$.

1. *The number of parking functions* $\mathbf{a}$ *of length n with* $\mathrm{Des}(\mathbf{a}) = S$ *is equal to* $\beta_{\mathrm{NC}_{n+1}}([n-1]-S)$.
2. *The number of parking functions* $\mathbf{a}$ *of length n satisfying* $S \subseteq \mathrm{Des}(\mathbf{a})$ *is equal to* $\alpha_{\mathrm{NC}_{n+1}}([n-1]-S)$.

The value of $\alpha_{\mathrm{NC}_{n+1}}(T)$ for $T \subseteq [n-1]$ is computed in Theorem 3.2 of [26]. Assume $T = \{t_1 < t_2 < \cdots < t_r\}$. Set $t_0 = 0$, $t_{r+1} = n$, and $\delta_i = t_i - t_{i-1}$ for $1 \leq i \leq r+1$. Then

$$\alpha_{\mathrm{NC}_{n+1}}(T) = \frac{1}{n}\prod_{i=1}^{r+1}\binom{n+1}{\delta_i}.$$

Theorem 1.20 is used by Kim and Seo [50] to study the minimal transitive factorizations for permutations of cycle type (n) and $(1,n-1)$. Given a permutation σ in $\mathfrak{S}_n$, the *minimal transitive factorizations* of σ is a set of $n+\ell-2$ transpositions that generate the full symmetric group such that their product is σ, where ℓ is the number of cycles of σ. If the permutation σ is of cycle type $\lambda = (\lambda_1,\dots,\lambda_\ell)$, then the number of minimal transitive factorizations of σ is

$$(n+\ell-2)!n^{\ell-3}\prod_{i=1}^{\ell}\frac{\lambda_i^{\lambda_i}}{(\lambda_i-1)!}, \tag{1.15}$$

a formula originally suggested by Hurwitz but proved by Goulden and Jackson [41], and Bousquet-Mélou and Schaeffer [16]. In particular, when σ is of cycle type (n) or $(1,n-1)$, Hurwitz's formula reduces to n^{n-2} and $(n-1)^n$. Kim and Seo [50] presented combinatorial proofs for these two cases using an interplay between the minimal transitive factorizations, cycle chord diagrams, Stanley's labeling of maximal chains of NC_n, and parking functions.

1.3.3 Hyperplane arrangements

A (real) hyperplane arrangement is a discrete set of affine hyperplanes in $\mathbb{R}^n$. An especially important arrangement is the *braid arrangement* $\mathscr{B}_n$, which consists of all the hyperplanes $x_i - x_j = 0$ for $1 \le i < j \le n$. For a hyperplane arrangement $\mathscr{A}$, if we remove the union of the hyperplanes from $\mathbb{R}^n$, then we obtain a disjoint union of open cells, called the *regions* of $\mathscr{A}$. Fix a region R_0 of $\mathscr{A}$ and call it the *base region*. Given a region R of $\mathscr{A}$, let $d(R)$, the *distance* of R, be the number of hyperplanes H in $\mathscr{A}$ that separate R_0 from R, i.e., R_0 and R lie in different sides of H. Define the *distance enumerator* of $\mathscr{A}$ (with respect to R_0) to be the generating function

$$D_{\mathscr{A}}(q) = \sum_R q^{d(R)},$$

where R ranges over all regions of $\mathscr{A}$. For finite $\mathscr{A}$, $D_{\mathscr{A}}(q)$ is a polynomial in q, and $D_{\mathscr{A}}(1)$ is the number of regions of the hyperplane arrangement $\mathscr{A}$.

As an example, consider the braid arrangement $\mathscr{B}_n$ which has $n!$ regions, each corresponds to a permutation of $\mathfrak{S}_n$. It is natural to let R_0 be the region defined by the conditions $x_1 > x_2 > \cdots > x_n$. There is an elegant way of labeling the regions of $\mathscr{B}_n$ by integer sequences that leads to the formula of $D_{\mathscr{B}_n}(q)$. Let $e_i \in \mathbb{N}^n$ be the vector with a 1 in the i-th coordinate and 0s elsewhere. First label the base region R_0 by $\lambda(R_0) = (0,\dots,0)$. Suppose now that R has been labeled, and that R' is an unlabeled region which is separated from R by a unique hyperplane $x_i = x_j$, where $i < j$. Then set

$$\lambda(R') = \lambda(R) + e_i.$$

It is easy to check that this labeling is well-defined, independent of the order in which the regions are labeled, and the labels of regions of $\mathscr{B}_n$ are the sequences $(c_1,\dots,c_n)$ such that $0 \le c_i \le n-i$. Hence

$$D_{\mathscr{B}_n}(q) = (1+q)(1+q+q^2)\cdots(1+q+\cdots+q^{n-1}),$$

the standard q-analog of $n!$.

A deformation of the braid arrangement is the Shi arrangement $\mathscr{S}_n$ consisting of hyperplanes $x_i - x_j = 0, 1$ for $1 \le i < j \le n$. This deformation was first considered by Shi [74] in his investigation of the affine Weyl group $\tilde{A}_n$. Shi showed by group-theoretic techniques that the number of regions of Shi arrangement is $(n+1)^{n-1}$. Pak and Stanley [79] gave a bijective proof by constructing a labeling of $\mathscr{S}_n$ that is analogous to that of $\mathscr{B}_n$.

A labeling of $\mathscr{S}_n$.
Let the base region R_0 be defined by $x_1 > x_2 > \cdots > x_n$ and $x_1 - x_n < 1$. Note that this is the unique region of $\mathscr{S}_n$ contained between all pairs of parallel hyperplanes. Label R_0 by the n-tuple $(0, 0, \ldots, 0)$. Now suppose a region R is labeled, and R' is an unlabeled region which is separated from R by a hyperplane H of $\mathscr{S}_n$. Then define

$$\lambda(R') = \begin{cases} \lambda(R) + e_i, & \text{if } H \text{ is given by } x_i - x_j = 0 \text{ with } i < j \\ \lambda(R) + e_j, & \text{if } H \text{ is given by } x_i - x_j = 1 \text{ with } i < j. \end{cases}$$

The labeling λ is independent of the order of the hyperplanes, hence well-defined. In addition, if $\lambda(R) = (a_1, \ldots, a_n)$, then $d(R) = a_1 + \cdots + a_n$, that is, there are $a_1 + \cdots + a_n$ hyperplanes of $\mathscr{S}_n$ that separate R and R_0.

Figure 1.7 shows the labeling of $\mathscr{S}_3$, where the hyperplanes are projected to $x_1 + x_2 + x_3 = 0$.

Theorem 1.23 *The labeling λ defined above is a bijection from the regions of $\mathscr{S}_n$ to the set $\mathscr{PK}_n$ of all parking functions of length n. Consequently, the number of regions R for which i hyperplanes separate R from R_0 is equal to the number of parking functions of length n with displacement $D(\mathbf{a}) = \binom{n}{2} - i$.*

In addition to Theorem 1.23, Stanley [79] generalized the labeling of $\mathscr{S}_n$ to the extended Shi arrangement, and established connections between the extended Shi arrangement and other combinatorial subjects, in particular, generalized parking functions and rooted k-forests, (c.f. 1.4.4).

Athanasiadis and Linusson [5] constructed another simple bijection between the regions of the Shi arrangement and the set of parking functions. Their bijection can be generalized to any subarrangement of $\mathscr{S}_n$ containing the hyperplanes $x_i - x_j = 0$ and to the extended Shi arrangements. It also implies that the number of relatively bounded regions of $\mathscr{S}_n$ is $(n-1)^{n-1}$, where a region is relatively bounded if its intersection with the hyperplane $x_1 + \cdots + x_n = 0$ is bounded as a subset of Euclidean space. Athanasiadis and Linusson's bijection maps the relatively bounded regions of $\mathscr{S}_n$ to the *prime parking functions* of length n, a concept due to Gessel. Under our notation a *prime parking function* of length n is a sequence $(a_1, \ldots, a_n)$ of non-negative integers such that for all $1 \le j \le n-1$, the cardinality of the set $\{a_i : a_i < j\}$ is at least $j+1$.

Armstrong [2] considered the regions of the Shi arrangement as antichains or ideals in the poset of positive roots, and mapped the regions bijectively to certain labeled Dyck paths. The same labeled Dyck paths are commonly used in the study of

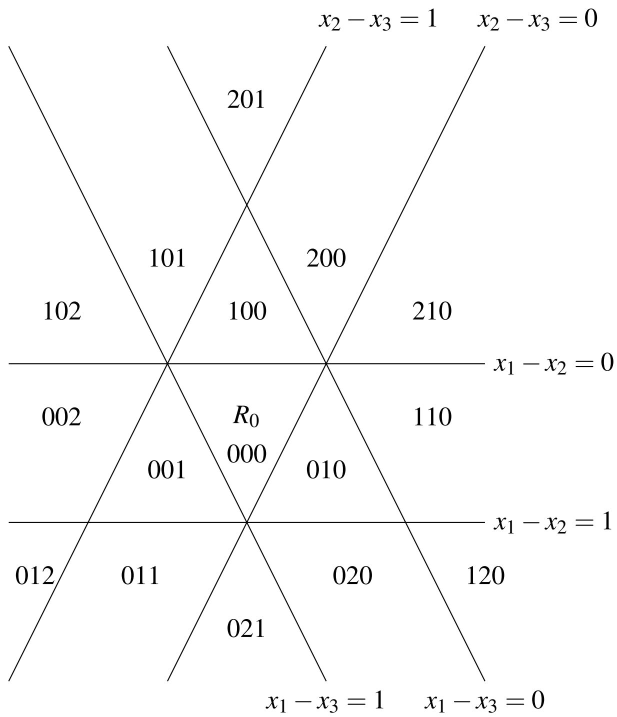

Figure 1.7
The labeling $\lambda(R)$ for the Shi arrangement $\mathscr{S}_3$.

diagonal harmonics to encode parking functions, (c.f. Section 1.6). Combining these two results one gets essentially the same bijection of Athanasiadis and Linusson's. There is some further elaboration of this labeling scheme in [4], which reveals the relations between the Shi arrangement and other hyperplane arrangements, as well as the relations among various statistics of parking functions.

1.3.4 Allowable input-output pairs in a priority queue

A *priority queue* is an abstract data type equipped with the operations INSERT and DELETEMIN. Let $\sigma = \sigma_1\sigma_2\ldots\sigma_n$ be a permutation of $[n]$. Each INSERT operation will insert the next element of σ into the queue, and each DELEDEMIN operation will remove the minimal element of the queue and place it in the output stream. A sequence of n INSERT and n DELETEMIN is an allowable sequence if any initial subsequence contains at least as many INSERT's as DELETEMIN's. The application of an allowable sequence to a permutation σ is called a priority queue computation. Assume the outcome is τ. Then (σ, τ) is called an allowable pair of permutations in a priority queue.

For example, when $n = 3$, the allowable sequences are *IDIDID*, *IDIIDD*, *IIDIDD*, *IIDDID*, *IIIDDD*, where *I* stands for INSERT and *D* for DELETEMIN. There are 16 allowable pairs of permutations, as listed below.

(123, 123)	(132, 132)	(132, 123)	(213, 213)
(213, 123)	(231, 231)	(231, 213)	(231, 123)
(312, 312)	(312, 123)	(312, 132)	(321, 321)
(321, 312)	(321, 213)	(321, 231)	(321, 123)

Let $\mathcal{Q}_n$ be the set of all allowable pairs of permutations on $[n]$. Atkinson and Thiyagarajah [6] found that the number of allowable pairs in $\mathcal{Q}_n$ is $(n+1)^{n-1}$. Gilbey and Kalikow [39] constructed a bijection between $\mathcal{Q}_n$ and $\mathcal{PK}_n$ that for each allowable pairs (σ, τ), the corresponding parking function has τ as output, that is, the car C_{τ_i} is parked at the space $i-1$ under the parking rules. In other words, τ is the permutation $\mathbf{q}$ as defined in Knuth's bijection ϕ described in Section 1.2.2. In addition, the bijection of Gilbey and Kalikow's has the extra property of preserving the set of breakpoints. For a parking function $\mathbf{a} = (a_1, \ldots, a_n)$, an integer $b \in \{0, 1, \ldots, n\}$ is a breakpoint if and only if $|\{i : p(i) \le b-1| = b$, while for an allowable pair (σ, τ), $b \in [0, n]$ is a breakpoint if and only if $\{\sigma_1, \ldots, \sigma_b\} = \{\tau_1, \ldots, \tau_b\}$. Note that under these definitions, 0 and n are always breakpoints of and $\mathbf{a} \in \mathcal{PK}_n$ and any $(\sigma, \tau) \in \mathcal{Q}_n$.

1.3.5 Two variations of parking functions

We conclude this section with two variations of parking functions.

The first variation is the defective parking functions considered by Cameron, Johannsen, Prellberg, and Schweitzer [17]. Suppose that there are m drivers entering a one-way street with n parking spaces, each with a preferred parking space. The parking procedure is the same as before. If k drivers fail to park, the preference sequence is called a *defective parking function of defect k*, whose number is denoted by $\mathrm{cp}(n,m,k)$. The original parking functions correspond to the case that $m = n$ and $k = 0$.

For the case $m < n$ and $k = 0$, Pollak's argument can be adapted to give $\mathrm{cp}(n,m,0) = (n+1-m)(n+1)^{m-1}$. The formula for general case is much harder. Cameron et al. established a recurrence relation for $\mathrm{cp}(n,m,k)$, and expressed it as an equation for a three-variable generating function. Solving the equation by using the kernel method, they obtained the following result.

Theorem 1.24 *The number of defective parking functions of defect k is given by*

$$\mathrm{cp}(n,m,k) = S(n,m,k) - S(n,m,k+1),$$

where $S(n,m,k)$ is the number of car parking preferences of m cars on n spaces, such that at least k cars do not find a parking space. The values of $S(n,m,k)$ can be computed explicitly by

$$S(n,m,k) = \begin{cases} n^m & \text{if } k \le m-n, \\ \sum_{i=0}^{m-k} \binom{m}{i} (n-m+k)(n-m+k+i)^{i-1}(m-k-i)^{m-i} & \text{otherwise.} \end{cases}$$

These formulas were also derived by Pitman and Stanley [67] using the volume polynomial of a polytope, and by Yan [86] with combinatorial means.

Cameron et al. investigated the asymptotic behavior of defective parking functions. They proved that if $m = n + \lfloor y\sqrt{n} \rfloor$, then the limiting probability of at most $\lfloor x\sqrt{n} \rfloor$ drivers failing to park is

$$\lim_{n\to\infty} \frac{1}{n^m} \sum_{k=0}^{\lfloor x\sqrt{n} \rfloor} \mathrm{cp}(n,m,k) = \begin{cases} 1 - e^{-2x(x-y)} & \text{if } x > y, \\ 0 & \text{otherwise.} \end{cases}$$

This limit distribution implies that if $m < n + k$ for a fixed constant k, then

$$\frac{\mathrm{cp}(n,m,k)}{n^m} \sim \frac{2}{n} \cdot (2k - m + n) \cdot e^{-2k(k+n-m)/n}.$$

Another problem for defective parking functions is to find the limiting probability that all parking spaces are occupied. For $m < n$ this probability is clearly 0. It turns out that when m is linear in n, that is, $m = \lfloor cn \rfloor$ for a constant $c > 0$ and $k = m - n$,

$$\lim_{n\to\infty} \frac{\mathrm{cp}(n,m,k)}{n^m} = \begin{cases} 0 & \text{if } c \le 1, \\ 1 - e^{-c} \cdot \sum_{i\ge 1} \frac{(ci/e^c)^{i-1}}{i!} & \text{if } c > 1. \end{cases} \tag{1.16}$$

Formula (1.16) is proved in Theorem 10 of [17] and also in [77] by Spencer and Yan who used the Galton-Watson branching process to investigate the connections between parking functions and random labeled trees. The approach of [77] was further extended by Chassaing and Marckert [20] to obtain tight bounds for the moments of the width of rooted labeled trees.

The second variation was given by Zara [88] who described an interesting parking problem. Assume that in the middle of the well-known parking procedure, $n - k$ spaces are already taken, and the spaces $\mathbf{q} = \{q_1 < q_2 < \cdots < q_k\}$ are still available. Some other k cars want to take these spaces; they enter the street and start advancing. When they are in front of spaces $\mathbf{p} = \{p_1 < p_2 < \cdots < p_k\}$, the lights go off. The cars can only advance towards the end of the street and park in an unoccupied space. Will all of them be able to find parking spaces?

It is clear that all the cars can park if and only of $p_i \le q_i$ for all $1 \le i \le k$. If this condition is satisfied, we say that $\mathbf{p} \preceq \mathbf{q}$ and call $\mathbf{p}, \mathbf{q}$ an initial condition. Given an initial condition $\mathbf{p}, \mathbf{q}$, a possible final arrangement is given by a permutation $\tau \in \mathfrak{S}_k$: for each $i = 1, \ldots, k$, the car at p_i takes the space $q_{\tau(i)}$. A permutation τ is attainable from $\mathbf{p}, \mathbf{q}$ if τ can appear as the permutation associated with a final arrangement when the initial condition is $\mathbf{p}, \mathbf{q}$. In particular, if everyone plays safe and takes the first available space, just as what is defined in the notion of parking functions, the resulting τ is called the *safe permutation* for $\mathbf{p}, \mathbf{q}$.

Zara gave a complete characterization of safe permutations: if τ is a safe permutation, then it avoids pattern 231. Conversely, a 231-avoiding permutation τ can be realized as the safe permutation for some initial conditions if and only if $\mathrm{is}(\tau)$ is at least $2k - n$, where $\mathrm{is}(\tau)$ is the maximal length of the increasing subsequences of τ.

There is a surprising connection between the safe permutations and the paths in the Johnson graph $J(n,k)$, which is the graph whose vertices are the k-element subsets of $[n]$, and whose edges are (V_1,V_2) such that $V_1 \cap V_2$ has size $k-1$.

1.4 Generalized Parking Functions

1.4.1 u-parking functions

For a finite sequence $(x_1,x_2,\dots,x_n)$ of real numbers, rearrange the terms in non-decreasing order as $x_{(1)} \le x_{(2)} \le \cdots \le x_{(n)}$. The term $x_{(i)}$ is the i-th *order statistic* of the sequence $\mathbf{x}$. Let $\mathbf{u}$ be a non-decreasing sequence $(u_1,u_2,u_3,\dots)$ of positive integers. A $\mathbf{u}$-parking function of length n is a sequence $(x_1,x_2,\dots,x_n)$ satisfying $0 \le x_{(i)} < u_i$. The parking functions studied in previous sections correspond to the case $\mathbf{u} = (1,2,\dots,n)$, and are usually referred as *ordinary parking functions* in enumeration literature, or as *classical parking functions* in diagonal harmonic research (c.f. Section 1.6). However, in this paper we shall reserve the words ”classical parking functions“ to the case that $\mathbf{u}$ is an arithmetic progression (c.f. Section 1.4.4).

Let $\mathscr{PK}_n(\mathbf{u})$ be the set of $\mathbf{u}$-parking functions of length n, and $PK_n(\mathbf{u})$ the number of elements in $\mathscr{PK}_n(\mathbf{u})$. Clearly $PK_n(\mathbf{u})$ is a function of $u_1,\dots,u_n$. Less obvious is the fact that it is a homogeneous polynomial of $u_1,u_2,\dots,u_n$. Furthermore, there is a determinant formula for $PK_n(\mathbf{u})$.

Theorem 1.25 *The number $PK_n(\mathbf{u})$ of $\mathbf{u}$-parking functions of length n equals $n!\det(D)$, where D is the $n\times n$ matrix with ij-th entry equal to*

$$\frac{u_i^{j-i+1}}{(j-i+1)!},$$

if $j-i+1 \ge 0$ and 0 otherwise.

Theorem 1.25 is the discrete analog of a formula of Steck [82] for the cumulative distribution function of the random vector for order statistics of n independent random variables with uniform distribution on an interval. However, Steck's formula is for sequences of real values. To use it one needs to establish a connection between real sequences and integer sequences. One way to do it is to use the parking polytope introduced in Section 1.4.2. In Section 1.4.3 we provide another proof of Theorem 1.25 via the theory of Gončarov polynomials.

The determinant formula in Theorem 1.25, while giving a solution for $PK_n(\mathbf{u})$, is not easy to compute for a general sequence $\mathbf{u}$. An easy case known is when $\mathbf{u}$ is an arithmetic progression, i.e., $u_i = a+(i-1)b$ for some positive integers a,b. In that case $PK_n(a,a+b,\dots,a+(n-1)b) = a(a+nb)^{n-1}$, and the corresponding parking functions are called *classical parking functions*. Many enumerative results and combinatorial correspondences described in the previous two sections can be extended to classical parking functions, which we will discuss in Section 1.4.4.

For a sequence $\mathbf{u}$, let $\Delta(\mathbf{u})$ be the difference sequence $(u_1, u_2 - u_1, u_3 - u_2, \dots)$. Besides classical parking functions, the only $PK_n(\mathbf{u})$ that are computed explicitly are the following two cases.

Theorem 1.26

1. *If*

$$\Delta(\mathbf{u}) = (a, \overbrace{b, \dots, b}^{n-m-1}, \overbrace{c, \dots, c}^{m-1}, d),$$

then

$$PK_n(\mathbf{u}) = a \sum_{j=0}^{m} \binom{n}{j} (m+1-j)(c-b) \Big[a + (n-j)\Big]^{n-j-1}$$
$$\cdot \Big[\big((m+1)c - (m+1-j)b\big)^{j-1} + j(d-c)\big(mc - (m+1-j)b\big)^{j-2}\Big]. \quad (1.17)$$

2. *If*

$$\Delta(\mathbf{u}) = (a, \overbrace{b, \dots, b}^{n-m-1}, d, \overbrace{c, \dots, c}^{m-1}),$$

then

$$PK_n(\mathbf{u}) = a \sum_{j=0}^{m} \binom{n}{j} \big[d + (m-j)c - (m+1-j)\big]$$
$$\cdot \big[d + mc - (m+1-j)b\big]^{j-1} \big[a + (n-j)b\big]^{n-j-1}. \quad (1.18)$$

Theorem 1.26 was proved in [86] by a combinatorial decomposition that partitions every integer sequence into two subsequences: a "maximum" parking function and a subsequence consisting of terms of higher values. Formula (1.18) with $c = 0$ was also given in [67] by connecting it to the empirical cumulative distributive function based on a sample of n independent uniform $(0,1)$ variables crossing an arbitrary line through the unit square. Note that when $\Delta(\mathbf{u}) = (n - (m-k) + 1, 1, \dots, 1, 0, \dots, 0)$, $PK_n(\mathbf{u})$ is exactly the number of defective parking functions with m drivers, n parking spaces, and at most k defects, as described in Section 1.3.5.

In Section 1.4.2 we introduce a combinatorial representation of a $\mathbf{u}$-parking function of length n: it equals the volume of certain polytope in $\mathbb{R}^n$ which admits a number of interpretations, in terms of empirical distributions, plane partitions, and polytopal subdivisions. Then we present in Subsection 1.4.3 a special sequence of polynomials, namely, Gončarov polynomials, which form a natural basis for working with $\mathbf{u}$-parking functions. The connection between Gončarov polynomials and $\mathbf{u}$-parking functions was established by extending the decomposition of [86]. Many properties of $\mathbf{u}$-parking functions can be derived from the theory of parking polytopes and the Gončarov polynomials. In particular, there are various formulas that allow us to compute the value of $PK_n(\mathbf{u})$ efficiently.

1.4.2 A parking polytope

In [67] Pitman and Stanley introduced an n-dimensional polytope Π_n whose volume defines a polynomial that has many combinatorial interpretations. Actually the volume polynomial of Π_n is a variant form of **u**-parking functions. We present some basic properties of the polytope Π_n and its volume polynomial.

Let

$$\Pi_n(\mathbf{x}) := \{\mathbf{y} \in \mathbb{R}^n \ : \ y_i \geq 0 \text{ and } \sum_{i=1}^{j} y_i \leq \sum_{i=1}^{j} x_i \text{ for all } 1 \leq j \leq n\}$$

for arbitrary $\mathbf{x} = (x_1, \ldots, x_n)$ with $x_i > 0$ for all i. The n-dimensional volume

$$V_n(\mathbf{x}) = Vol(\Pi_n(\mathbf{x}))$$

is a homogeneous polynomial of degree n in the variables $x_1, \ldots, x_n$, which is called the *volume polynomial*. For example, volume polynomials of small indices are

$$\begin{aligned} V_1(\mathbf{x}) &= x_1, \\ V_2(\mathbf{x}) &= x_1x_2 + \frac{1}{2}x_1^2, \\ V_3(\mathbf{x}) &= x_1x_2x_3 + \frac{1}{2}x_1^2x_2 + \frac{1}{2}x_1x_2^2 + \frac{1}{2}x_1^2x_3 + \frac{1}{6}x_1^3. \end{aligned}$$

The volume polynomial $V_n(\mathbf{x})$ is closely related to the order statistics of n independent random variables. Let $U_1, U_2, \ldots, U_n$ be n independent random variables uniformly distributed in $(0,1)$, and $\{U_{(i)} : \ i \leq i \leq n)$ be the order statistics of $U_1, U_2, \ldots, U_n$. Because the random vectors $(U_{(i)} : 1 \leq i \leq n)$ and $(1 - U_{(n+1-i)} : \ 1 \leq i \leq n)$ have the same uniform distribution with constant density $n!$ on the simplex

$$\{\mathbf{u} \in \mathbf{R}^n : 0 \leq u_1 \leq \cdots \leq u_n \leq 1\}, \tag{1.19}$$

one obtains that for arbitrary vectors **u** and **r** in this simplex

$$\Pr(U_{(i)} \leq u_j \text{ for all } 1 \leq j \leq n) = n! V_n(x_1, \ldots, x_n) \tag{1.20}$$

where $(x_1, \ldots, x_n) = \Delta(\mathbf{u})$, or equivalently, $u_i = \sum_{j=1}^{i} x_j$, and

$$\Pr(U_{(i)} \geq r_i \text{ for all } 1 \leq i \leq n) = n! V_n(x_1, \ldots, x_n) \tag{1.21}$$

where $x_j = r_{n+2-j} - r_{n+1-j}$ (with the convention that $r_{n+1} = 1$). Formulas (1.20) and (1.21) allow us to compute $V_n(\mathbf{x})$ using the probability theory.

Theorem 1.27 *For a positive integer n,*

$$V_n(\mathbf{x}) = \sum_{\mathbf{k} \in K_n} \prod_{i=1}^{n} \frac{x_i^{k_i}}{k_i!} = \frac{1}{n!} \sum_{\mathbf{k} \in K_n} \binom{n}{k_1, \ldots, k_n} x_1^{k_1} \cdots x_n^{k_n}, \tag{1.22}$$

where K_n is the set of balanced vectors of length n, that is,

$$K_n = \{\mathbf{k} \in \mathbf{N}^n \ : \ \sum_{i=1}^{j} k_i \geq j \textit{ for all } 1 \leq i \leq n-1 \textit{ and } \sum_{i=1}^{n} k_i = n\}.$$

Proof. Let $u_i = \sum_{j=1}^{i} x_j$. By homogeneity of V_n, it suffices to prove Formula (1.22) with $u_n \le 1$. Fix $\mathbf{u}$ and consider the probability in (1.20). For $1 \le i \le n+1$ let N_i be the number of U_i that lying in the interval $(u_{i-1}, u_i]$, with the conventions that $u_0 = 0$ and $u_{n+1} = 1$.

Since U_i are independent and uniformly distributed in $(0,1)$, the random vector $(N_i : 1 \le k \le n+1)$ has the *multinomial distribution* with parameters n and $(x_1, \ldots, x_n, x_{n+1})$, where $x_{n+1} = 1 - u_n = 1 - \sum_{i=1}^{n} x_i$. In other words, for any sequence $(k_1, \ldots, k_{n+1})$ with $\sum_{i=1}^{n+1} = n$, we have

$$\Pr[(N_1, \ldots, N_{n+1}) = (k_1, \ldots, k_{n+1})] = \binom{n}{k_1, \ldots, k_{n+1}} \prod_{i=1}^{n+1} x_i^{k_i} = n! \prod_{i=1}^{n+1} \frac{x_i^{k_i}}{k_i!}.$$

Note that N_i also equals the number of $U_{(i)}$'s that lying in $(u_{i-1}, u_i]$. Hence $U_{(i)} \le u_i$ if and only if $\sum_{i=1}^{i} N_j \ge i$. It follows

$$\begin{aligned} & \Pr(U_{(i)} \le u_i \text{ for all } 1 \le i \le n) \\ = \quad & \Pr(\sum_{i=1}^{i} N_j \ge i \text{ for all } 1 \le i \le n) \\ = \quad & \sum_{\mathbf{k} \in K_n} \Pr((N_1, \ldots N_n, N_{n+1}) = (k_1, \ldots, k_n, 0)) \\ = \quad & n! \sum_{\mathbf{k} \in K_n} \prod_{i=1}^{n} \frac{x_i^{k_i}}{k_i!}. \end{aligned}$$

Combining the above equation with (1.20) we prove Theorem 1.27. □

From the above proof follow two more probabilistic interpretations of $V_n(\mathbf{x})$.

Corollary 1.28 *Let $(N_i, 1 \le i \le n+1)$ be a random vector with multinomial distribution with parameters n and $(p_1, \ldots, p_{n+1})$, as if N_i is the number of times i appears in a sequence of n independent trials with probability p_i of getting i on each trial for $1 \le i \le n+1$, where $\sum_{i=1}^{n+1} p_i = 1$. Then*

$$\Pr\left(\sum_{j=1}^{i} N_j \ge i \text{ for all } 1 \le i \le n\right) = n! V_n(p_1, p_2, \ldots, p_n),$$

and

$$\Pr\left(\sum_{j=1}^{i} N_j < i \text{ for all } 1 \le i \le n\right) = n! V_n(p_{n+1}, p_n, \ldots, p_2).$$

The relation between $V_n(\mathbf{x})$ and the generalized parking functions is given by the following theorem.

Theorem 1.29 *Assume that $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{N}^n$. Let $u_j = \sum_{i=1}^{j} x_i$, i.e., $\mathbf{x} = \Delta(\mathbf{u})$. Then*

$$PK_n(\mathbf{u}) = \sum_{\mathbf{a} \in \mathscr{PK}_n} x_{a_1+1} \cdots x_{a_n+1} = n! V_n(\mathbf{x}). \tag{1.23}$$

Proof. Given an ordinary parking function $\mathbf{a} = (a_1, \ldots, a_n)$, replace each $i-1$ by an integer in the set $\{x_1 + \cdots + x_{i-1}, \ldots, x_1 + \cdots + x_i - 1\}$. The number of ways to do this is given in the middle expression in (1.23), and every $\mathbf{u}$-parking function is obtained exactly once in this way. This gives the first equality. The second equality follows from Theorem 1.27 and 1.9, as every ordinary parking function $\mathbf{a}$ is obtained by a balanced vector $\mathbf{k} \in K_n$ which determines that the terms of $\mathbf{a}$ contain exactly k_i $(i-1)$'s, and a compatible permutation that determines how the terms are arranged. Given a balanced vector $\mathbf{k} \in K_n$, it is clear that there are $\binom{n}{k_1,\ldots,k_n}$ permutations that are compatible with $\mathbf{k}$. □

In particular, Formula (1.22) gives an explicit way to compute $PK_n(\mathbf{u})$, which is much easier than the determinant formula in Theorem 1.25.

An interesting special case of Theorem 1.29 is when we take $x_i = q^{i-1}$. In this case we have

$$n!V_n(1,q,q^2,\ldots,q^{n-1}) = \sum_{\alpha \in PK_n} q^{a_1+\cdots+a_n} = q^{\binom{n}{2}} P_n(1/q) = q^{\binom{n}{2}} I_n(1/q),$$

where $P_n(q)$ is the displacement enumerator of ordinary parking functions, and $I_n(q)$ is the inversion enumerator of labeled trees.

The polytope $\Pi_n(\mathbf{x})$ has many interesting properties. For example, it admits a subdivision into a collection of n-dimensional chambers, with the volume of each chamber corresponding to a term of the volume polynomial. It also relates to plane partitions, rooted binary trees, and another polytope called *associahedron*. More combinatorial properties of $\Pi_i(\mathbf{x})$ are discussed in [67].

1.4.3 Theory of Gončarov polynomials

In this section we describe a polynomial sequence that forms a natural basis for working with $\mathbf{u}$-parking functions. The involved polynomials are called Gončarov polynomials, which arose from the Gončarov interpolation problem in numerical analysis.

> [**Gončarov Interpolation**] Given two sequences of real or complex numbers $a_0, a_1, \ldots, a_n$ and $b_0, b_1, \ldots, b_n$, find a polynomial $p(x)$ of degree n such that for each $i, 0 \le i \le n$, the ith derivative $p^{(i)}(x)$ evaluated at a_i equals b_i.

Gončarov polynomials are the basis of solutions to the Gončarov interpolation problem and correspond to the cases that all b_i but one are zero. To state their properties, we start with a brief introduction of the theory of sequences of polynomial biorthogonal to a sequence of linear functionals. The details can be found in [60].

Let $\mathscr{P}$ be the vector space of all polynomials in the variable x over a field F of characteristic zero. Let $D : \mathscr{P} \to \mathscr{P}$ be the differentiation operator. For a scalar a in the field F, let

$$\varepsilon(a) : \mathscr{P} \to F,\ p(x) \mapsto p(a)$$

be the linear functional which evaluates $p(x)$ at a.

Let $\varphi_s(D), s = 0, 1, 2, \ldots$ be a sequence of linear operators on $\mathscr{P}$ of the form

$$\varphi_s(D) = D^s \sum_{r=0}^{\infty} b_{sr} D^r,$$

where the coefficients b_{s0} are assumed to be non-zero. There exists a unique sequence $p_n(x), n = 0, 1, 2, \ldots$ of polynomials such that $p_n(x)$ has degree n and

$$\varepsilon(0)\varphi_s(D)p_n(x) = n!\delta_{sn},$$

where δ_{sn} is the Kronecker delta.

The polynomial sequence $p_n(x)$ is said to be *biorthogonal* to the sequence $\varphi_s(D)$ of operators, or, the sequence $\varepsilon(0)\varphi_s(D)$ of linear functionals. Using Cramer's rule to solve the linear system and Laplace's expansion to group the results, we can express $p_n(x)$ by the the following *determinantal formula*:

$$p_n(x) = \frac{n!}{b_{00}b_{10}\cdots b_{n0}} \begin{vmatrix} b_{00} & b_{01} & b_{02} & \ldots & b_{0,n-1} & b_{0n} \\ 0 & b_{10} & b_{11} & \ldots & b_{1,n-2} & b_{1,n-1} \\ 0 & 0 & b_{20} & \ldots & b_{2,n-3} & b_{2,n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & b_{n-1,0} & b_{n-1,1} \\ 1 & x & x^2/2! & \ldots & x^{n-1}/(n-1)! & x^n/n! \end{vmatrix}.$$

Since $\{p_n(x)\}_{n=0}^{\infty}$ forms a basis of $\mathscr{P}$, any polynomial can be uniquely expressed as a linear combination of $p_n(x)$'s. Explicitly, we have the *expansion formula*: If $p(x)$ is a polynomial of degree n, then

$$p(x) = \sum_{i=0}^{n} \frac{d_i p_i(x)}{i!},$$

where $d_i = \varepsilon(0)\varphi_i(D)p(x)$. In particular,

$$x^n = \sum_{i=0}^{n} \frac{n! b_{i,n-i} p_i(x)}{i!},$$

which gives a *linear recursion* for $p_n(x)$. Equivalently, one can write the above equation in terms of formal power series equations, and obtain the *Appell relation*

$$e^{xt} = \sum_{n=0}^{\infty} \frac{p_n(x)\varphi_n(t)}{n!},$$

where $\varphi_n(t) = t^s \sum_{r=0}^{\infty} b_{sr} t^r$.

A special example of sequences of biorthogonal polynomials is the Gončarov polynomials. Let $(a_0, a_1, a_2, \ldots)$ be a sequence of numbers or variables called *nodes*. The sequence of *Gončarov polynomials*

$$g_n(x; a_0, a_1, \ldots, a_{n-1}),\ n = 0, 1, 2, \ldots$$

is the sequence of polynomials biorthogonal to the operators

$$\varphi_S(D) = D^s \sum_{r=0}^{\infty} \frac{a_s^r D^r}{r!} = \varepsilon(a_s) D^s.$$

As indicated by the notation, $g_n(x; a_0, a_1, \ldots, a_{n-1})$ depends only on the nodes $a_0, a_1, \ldots, a_{n-1}$. In particular, when all the a_i equal a, we have

$$g_n(x; a, a, \ldots, a) = (x-a)^n$$

and Gončarov interpolation is just expansion as a power series at $x = a$. When $a_0, a_1, a_2, \ldots$ form an arithmetic progression $a, a+b, a+2b, \ldots$, we get Abel polynomials

$$g_n(x; a, a+b, a+2b, \ldots, a+(n-1)b) = (x-a)(x-a-nb)^{n-1}.$$

Gončarov polynomials have many nice algebraic and analytic properties, which make them very useful in analysis and combinatorics. Next we list some basic properties, of which the first four follow from the theory of biorthogonal polynomials.

Theorem 1.30 *The Gončarov polynomials $g_n(x; a_0, a_1, \ldots, a_{n-1})$ have the following properties.*

1. *Determinant formula.*

$$g_n(x; a_0, a_1, \ldots, a_{n-1}) = n! \begin{vmatrix} 1 & a_0 & \frac{a_0^2}{2!} & \frac{a_0^3}{3!} & \cdots & \frac{a_0^{n-1}}{(n-1)!} & \frac{a_0^n}{n!} \\ 0 & 1 & a_1 & \frac{a_1^2}{2!} & \cdots & \frac{a_1^{n-2}}{(n-2)!} & \frac{a_1^{n-1}}{(n-1)!} \\ 0 & 0 & 1 & a_2 & \cdots & \frac{a_2^{n-3}}{(n-3)!} & \frac{a_2^{n-2}}{(n-2)!} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & 0 & \cdots & 1 & a_{n-1} \\ 1 & x & \frac{x^2}{2!} & \frac{x^3}{3!} & \cdots & \frac{x^{n-1}}{(n-1)!} & \frac{x^n}{n!} \end{vmatrix}.$$

2. *Expansion formula. If $p(x)$ is a polynomial of degree n, then*

$$p(x) = \sum_{i=0}^{n} \frac{\varepsilon(a_i) D^i p(x)}{i!} g_i(x; a_0, a_1, \ldots, a_{i-1}).$$

3. *Linear recurrence. Let $p(x) = x^n$ in the expansion formula, we have*

$$x^n = \sum_{i=0}^{n} \binom{n}{i} a_i^{n-i} g_i(x; a_0, a_1, \ldots, a_{i-1}). \tag{1.24}$$

4. *Appell relation.*

$$e^{xt} = \sum_{n=0}^{\infty} g_n(x; a_0, a_1, \ldots, a_{n-1}) \frac{t^n e^{a_n t}}{n!}.$$

5. *Differential relations. The Gončarov polynomials can be equivalently defined by the differential relations*

$$Dg_n(x;a_0,a_1,\ldots,a_{n-1}) = ng_{n-1}(x;a_1,a_2,\ldots,a_{n-1}),$$

with initial conditions

$$g_n(a_0;a_0,a_1,\ldots,a_{n-1}) = \delta_{0,n}.$$

6. *Integral relations.*

$$\begin{aligned} g_n(x;a_0,a_1,\ldots,a_{n-1}) &= n\int_{a_0}^{x} g_{n-1}(t;a_1,a_2,\ldots,a_{n-1})dt \\ &= n!\int_{a_0}^{x} dt_1 \int_{a_1}^{t_1} dt_2 \cdots \int_{a_{n-1}}^{t_{n-1}} dt_n. \end{aligned}$$

7. *Shift invariance formula.*

$$g_n(x+\xi;a_0+\xi,a_1+\xi,\ldots,a_{n-1}+\xi) = g_n(x;a_0,a_1,\ldots,a_{n-1}).$$

8. *Perturbation formula.*

$$\begin{aligned} & g_n(x;a_0,\ldots,a_{m-1},a_m+b_m,a_{m+1},\ldots,a_{n-1}) = g_n(x;a_0,\ldots a_{m-1},a_m,a_{m+1},\ldots,a_{n-1}) \\ - & \binom{n}{m} g_{n-m}(a_m+b_m;a_m,a_{m+1},\ldots,a_{n-1})g_m(x;a_0,a_1,\ldots,a_{m-1}). \end{aligned}$$

9. *Sheffer relation.*

$$g_n(x+y;a_0,\ldots,a_{n-1}) = \sum_{i=0}^{n} \binom{n}{i} g_{n-i}(y;a_i,\ldots,a_{n-1})x^i.$$

In particular,

$$g_n(x;a_0,\ldots,a_{n-1}) = \sum_{i=0}^{n} \binom{n}{i} g_{n-i}(0,a_i,\ldots,a_{n-1})x^i.$$

That is, coefficients of Gončarov polynomials are constant terms of (shifted) Gončarov polynomials.

It turns out that **u**-parking functions provide a combinatorial interpretation of Gončarov polynomials.

Theorem 1.31 *Let* $\mathbf{u} = (u_1,u_2,\ldots)$ *be a sequence of non-decreasing positive integers. Then we have*

$$\begin{aligned} PK_n(\mathbf{u}) = PK_n(u_1,u_2,\ldots,u_n) &= g_n(x;x-u_1,x-u_2,\ldots,x-u_n) \\ &= g_n(0;-u_1,-u_2,\ldots,-u_n) \\ &= (-1)^n g_n(0;u_1,u_2,\ldots,u_n). \end{aligned}$$

Proof. Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ be an integer sequence whose terms satisfy $0 \leq x_i < x$ for a positive integer x, and $(x_{(1)}, x_{(2)}, \ldots, x_{(n)})$ be the order statistics of $\mathbf{x}$. Let m be the maximum index such that

$$x_{(i)} < u_i \quad \text{for} \quad i = 1, 2, \ldots, m. \tag{1.25}$$

Then, the subsequence $(x_{i_1}, x_{i_2}, \ldots, x_{i_m})$ from which the sequence $(x_{(1)}, x_{(2)}, \ldots, x_{(m)})$ was obtained by rearrangement is a $\mathbf{u}$-parking function of length m. Furthermore, m is the maximum index satisfying condition (1.25) if and only if

$$x_{(n)} \geq x_{(n-1)} \geq \ldots \geq x_{(m+1)} \geq u_{m+1}.$$

Equivalently, the complementary subsequence $(x_{j_1}, x_{j_2}, \ldots, x_{j_{n-m}})$, obtained by deleting the subsequence $(x_{i_1}, x_{i_2}, \ldots, x_{i_m})$ from the original sequence, takes values in the interval $[u_{m+1}, x-1]$. Since the maximum index m and the set $\{i_1, i_2, \ldots, i_m\}$ are uniquely determined by the sequence $(x_1, x_2, \ldots, x_n)$, and any pair of subsequences satisfying the conditions in the theorem can be reassembled into a sequence in $[0, x-1]^n$, this decomposition yields a bijection, which leads to the equation

$$x^n = \sum_{m=0}^{n} \binom{n}{m} (x - u_{m+1})^{n-m} PK_m(u_1, u_2, \ldots, u_m). \tag{1.26}$$

Comparing the recursion (1.26) with the linear recursion (1.24) for Gončarov polynomials, we obtain

$$PK_n(u_1, u_2, \ldots, u_n) = g_n(x; x - u_1, x - u_2, \ldots, x - u_n).$$

By the shift invariance formula,

$$g_n(x; x - u_1, x - u_2, \ldots, x - u_n) = g_n(0; -u_1, -u_2, \ldots, -u_n).$$

Since the Gončarov polynomial $g_n(x; a_0, a_1, \ldots, a_{n-1})$ is a homogeneous polynomial of total degree n in $x, a_0, a_1, \ldots, a_{n-1}$, we have

$$g_n(0; -u_1, -u_2, \ldots, -u_n) = (-1)^n g_n(0; u_1, u_2, \ldots, u_n).$$

□

Any reasonable formula for Gončarov polynomials yields a reasonable formula for $\mathbf{u}$-parking functions. For example, when $u_i = a + (i-1)b$, we obtain the following special case.

Corollary 1.32

$$PK_n(a, a+b, a+2b, \ldots, a+(n-1)b) = a(a+nb)^{n-1}.$$

The homogeneity of Gončarov polynomials implies

Corollary 1.33

$$PK_n(bu_1, bu_2, \ldots, bu_n) = b^n PK_n(u_1, u_2, \ldots, u_n).$$

Using the determinant formula of Theorem 1.30, we get a proof of Theorem 1.25, the discrete analogue of Steck's formula. Formulas (1.17) and (1.18) of Theorem 1.26 can be proved by using the perturbation formula of Gončarov polynomials. In addition, one could compute the sum enumerator of **u**-parking functions by applying the properties of Gončarov polynomials and the decomposition of Theorem 1.31.

The *sum enumerator* $S_n(q;\mathbf{u})$ of the set of **u**-parking functions is the polynomial in q defined by

$$S_n(q;\mathbf{u}) = \sum_{(a_1,a_2,\ldots,a_n)} q^{a_1+a_2+\ldots+a_n}$$

where the sum ranges over all **u**-parking functions $(a_1,a_2,\ldots,a_n)$. The sum enumerator may be regarded as a "q-analogue" of $PK_n(\mathbf{u})$.

Theorem 1.34 *The sum enumerator $S_n(q;\mathbf{u})$ satisfies the equation*

$$(1+q+q^2+\ldots+q^{x-1})^n = \sum_{m=0}^{n} \binom{n}{m} (q^{u_{m+1}} + q^{u_{m+1}+1} + \ldots + q^{x-1})^{n-m} S_m(q;\mathbf{u}).$$

Proof. Since sum enumerators are multiplicative, the sum enumerator of integer sequences $(x_1,\ldots,x_n)$ with $0 \le x_i < x$ is

$$(1+q+q^2+\ldots+q^{x-1})^n.$$

For the same reason, the sum enumerator of sequences which are decomposed into a **u**-parking function of length m and a sequence in $[u_{m+1}, x-1]^{n-m}$ is

$$(q^{u_{m+1}} + q^{u_{m+1}+1} + \ldots + q^{x-1})^{n-m} S_m(q,\mathbf{u}).$$

The recursion now follows. □

Comparing the recursion in Theorem 1.34 with the linear recursion (1.24), we obtain

$$S_n(q;\mathbf{u}) = PK_n(1+q+\ldots+q^{u_1-1}, 1+q+\ldots+q^{u_2-1}, \ldots, 1+q+\ldots+q^{u_n-1}).$$

Using Theorem 1.31 and the shift invariance formula, we can express sum enumerators in terms of Gončarov polynomials, as

$$S_n(q;\mathbf{u}) = g_n\left(\frac{1}{1-q}; \frac{q^{u_1}}{1-q}, \frac{q^{u_2}}{1-q}, \ldots, \frac{q^{u_n}}{1-q}\right).$$

By homogeneity of Gončarov polynomials,

$$(1-q)^n S_n(q;\mathbf{u}) = g_n(1; q^{u_1}, q^{u_2}, \ldots, q^{u_n}).$$

Hence, sum enumerators satisfy the simpler linear recursion

$$1 = \sum_{m=0}^{n} \binom{n}{m} q^{u_{m+1}(n-m)} (1-q)^m S_m(q;\mathbf{u}). \tag{1.27}$$

They also satisfy the following Appell relation

$$\exp(t) = \sum_{n=0}^{\infty} (1-q)^n S_n(q;\mathbf{u}) \exp(q^{u_{n+1}} t) \frac{t^n}{n!}.$$

In the case of ordinary parking functions with $u_i = i$, we have

$$(1-q)^n S_n(q;1,2,\ldots,n) = g_n(1;q,q^2,\ldots,q^n),$$

and the sum enumerator is related to the displacement enumerator of ordinary parking functions by the equation

$$S_n(q;1,2,\ldots,n) = q^{\binom{n}{2}} P_n(1/q).$$

Hence (1.27) leads to a linear recurrence for $P_n(q)$:

$$1 = \sum_{m=0}^{n} q^{\binom{m+1}{2}-(m+1)n} \cdot (q-1)^m P_m(q).$$

More properties and computation of parking functions via Gončarov polynomials are given in [59, 60]. For example, [60] gives the generating functions for factorial moments of sums of **u**-parking functions, while the explicit formulas for the first and second factorial moments of sums of **u**-parking functions are given in [59]. Khare, Lorentz and Yan [49] studied multivariate Gončarov polynomials, and extended many algebraic and analytic properties of Gončarov polynomials to the multivariate case. They also established a connection between multivariate Gončarov polynomials and order statistics of integer sequences, which leads to a higher dimensional generalization of parking functions.

If one changes the differential operator D to the backward different operator Δ that maps a polynomial $p(x)$ to $\Delta p(x) = p(x) - p(x-1)$, then one gets the *difference Gončarov polynomials*, whose combinatorial counterpart is the set of lattice paths with a given right boundary. Theory of difference Gončarov polynomials is presented in [57].

1.4.4 Classical parking functions

A special class of **u**-parking functions is the classical parking functions, for which the entries of the vector **u** form an arithmetic progression. In particular, if $u_i = a + (i-1)b$ for some positive integers a and b, we will call the **u**-parking functions (a,b)-*parking functions*. Classical parking functions have a rich theory. Many combinatorial representations of ordinary parking functions can be generalized to this case, much in the same way as theory of Catalan structures (structures counted by the Catalan numbers $\frac{1}{n+1}\binom{2n}{n}$) generalizes to that of the Fuss-Catalan structures (structures counted by the Fuss-Catalan numbers, or k-Catalan numbers $\frac{1}{kn+1}\binom{nk+1}{n}$). In this section we give a brief summary of such generalizations.

Basic enumeration.

It is well-known that the number of (a,b)-parking functions is $a(a+nb)^{n-1}$.

Let $E_k(n;a,b)$ be the expected value of the k-th factorial moment of the sum of the terms in a uniform random (a,b)-parking function, that is,

$$E_k(n;a,b) = \frac{1}{a(a+nb)^{n-1}} \sum_{(x_1,x_2,\ldots,x_n)} (x_1+x_2+\ldots+x_n)_k,$$

where $(n)_k$ is the falling factorial $n(n-1)\cdots(n-k+1)$ and the sum ranges over all (a,b)-parking functions $(x_1,x_2,\ldots,x_n)$. Explicit formulas for $E_1(n;a,b)$ and $E_2(n;a,b)$ are computed by Kung and Yan in [58, 59].

Theorem 1.35 *The expected sums of terms in a random (a,b)-parking function of length n is*

$$E_1(n;a,b) = \frac{n(a+nb+1)}{2} - \frac{1}{2}\sum_{j=1}^{n}\binom{n}{j}\frac{j!b^j}{(a+nb)^{j-1}}.$$

Theorem 1.36 *The second factorial moment of the sum of terms of a random (a,b)-parking function of length n is*

$$\begin{aligned} &\tfrac{1}{4}n(n-1)(a+nb+1)^2 + \tfrac{1}{3}n(a+nb+1)(a+nb-1) \\ - &\frac{n(a+nb+1)}{2}\sum_{j=1}^{n}\binom{n}{j}\frac{j!b^j}{(a+nb)^{j-1}} + \sum_{j=1}^{n}\binom{n}{j}\frac{j!b^j}{(a+nb)^{j-1}}\left(\frac{b}{6}j^3 - \frac{a}{6}j + \frac{1}{2}\right). \end{aligned}$$

Kung and Yan also presented a general form for the higher moments of the expected sum of (a,b)-parking functions, see Theorem 1.1 of [58].

Rooted *b*-forests

The correspondence between ordinary parking functions and labeled trees can be extended to (a,b)-parking functions, where the labeled trees are replaces by forests of labeled trees with edge colors. Precisely, define a *rooted b-forest on* $[n]$ to be a rooted forest on the vertex set $[n]$ whose edges are colored with the colors $0,1,\ldots,b-1$. There is no further restriction on the possible coloring of the edges. Let $\mathscr{F}_n(a,b)$ be the set of all sequences $(T_1,T_2,\ldots,T_a)$ of length a such that (1) each T_i is a rooted b -forest, (2) T_i and T_j are disjoint if $i \neq j$, and (3) the union of the vertex sets of $T_1,\ldots,T_a$ is $[n]$.

Denote by $\mathscr{PK}_n((a,b))$ the set of (a,b)-parking functions of length n. Then there is a one-to-one correspondence between $\mathscr{PK}_n((a,b))$ and $\mathscr{F}_n(a,b)$. In fact, both of them can be mapped bijectively to the set $\mathscr{C}_n(a,b)$ that consists of pairs $(\vec{r},\sigma)$ such that

1. $\vec{r} = (r_0,r_1,\ldots,r_{a+(n-1)b-1}) \in \mathbb{N}^{a+(n-1)b}$ is (a,b)-balanced, that is,

$$\begin{aligned} &r_0+r_1+\cdots+r_{a+ib}-1 \geq i+1 \quad \text{for } i=0,1,\ldots,n-2, \\ &r_0+r_1+\cdots+r_{a+(n-1)b-1} = n. \end{aligned}$$

2. $\sigma \in \mathfrak{S}_n$ is compatible with $\vec{r}$, that is, the terms in the inverse σ^{-1} of σ is increasing on every interval of the indices $\{1+\sum_{i=1}^{k} r_i, 2+\sum_{i=1}^{k} r_i, \ldots, \sum_{i=1}^{k+1} r_i\}$ (if $r_{k+1} \neq 0$).

These results were proved in [87] by using the breadth-first search on rooted b-forests, which generalize Theorems 1.9 and 1.10. Eu, Fu and Lai [27, 28] applied the construction of [87] to enumerate (a,b)-parking functions by their leading terms, or with certain symmetric restrictions and periodic restrictions.

Inversions of b-forests and multicolored graphs

The notion of inversions can be extended to the set $\mathscr{F}_n(a,b)$ as follows: Let $F = (T_1, T_2, \ldots, T_a)$ be a sequence of rooted b-forests on $[n]$. Denote the color of an edge e by $\kappa(e)$. Define the (a,b)-inversion $\mathrm{inv}^{(a,b)}(F)$ by letting

$$\mathrm{inv}^{(a,b)}(F) = \mathrm{inv}(F) + \sum_{i=1}^{a} (i-1)|T_i| + \sum_{x\in[n]} \sum_{e\in K(x)} \kappa(e),$$

where $\mathrm{inv}(F)$ is the number of inversions of $T_1 \cup T_2 \cdots \cup T_a$ as an ordinary rooted forest, $K(x)$ is the set of edges lying between the vertex x and the root of the unique tree to which x belongs. Define the (a,b)-*inversion enumerator* $I_n^{(a,b)}(q)$ by

$$I_n^{(a,b)}(q) = \sum_{F\in\mathscr{F}_n(a,b)} q^{\mathrm{inv}^{(a,b)}(F)}.$$

For $a = 1$, $I_n^{(a,b)}(q) = I_n^{(1,b)}(q)$ is the b-inversion enumerator studied in [81, 85], and $I_n^{(1,1)}(q) = I_n(q)$ is the ordinary inversion enumerator of labeled trees.

For an (a,b)-parking function $\mathbf{a} = (a_1, \ldots, a_n)$, the (a,b)-*displacement* $D^{(a,b)}(\mathbf{a})$ is defined as

$$D^{(a,b)}(\mathbf{a}) = b\binom{n}{2} + an - \sum_i a_i,$$

and the (a,b)-*displacement enumerator* is

$$P_n^{(a,b)}(q) = \sum_{\mathbf{a}\in\mathscr{PK}_n((a,b))} q^{D^{(a,b)}(\mathbf{a})}.$$

Then we have

Theorem 1.37

$$I_n^{(a,b)}(q) = P_n^{(a,b)}(q).$$

Both of the above polynomials are related to another polynomial $C_n^{(a,b)}(q)$ that counts the number of edges in connected multigraphs. To wit, define a *multicolored* (a,b)-*graph* on $[n]$ to be a graph G with the vertex set $[n]$ such that

1. The edges of G are colored with colors $0, 1, \ldots, b-1$,

2. There are no loops or multiple edges of the same color in G. But G may have edges with the same endpoints but different colors, and

3. every vertex r is assigned with a subset $f(r)$ of $[a] = \{1, 2, \ldots, a\}$. We say that r is a *root* of G if $f(r) \neq \emptyset$.

4. For any subgraph H of G, define $R(H) = \sum_{r \in H} |f(r)|$ to be the number of roots in H, counting multiplicity. Every connected component G' of G has at least one root, i.e., $R(G') > 0$.

Denote by $e(G)$ the number of edges of G, and by $r(G) = \sum_r |f(r)|$ the number of roots of G. Let

$$C_n^{(a,b)}(q) = \sum_G q^{e(G)+r(G)-n}$$

where G ranges over all multicolored (a,b)-graphs on $[n]$. Set $C_0^{(a,b)}(q) = 1$ for all $a, b \in \mathbb{N}$. Then

Theorem 1.38

$$I_n^{(a,b)}(1+q) = P_n^{(a,b)}(1+q) = C_n^{(a,b)}(q).$$

Theorem 1.38 is proved in [87] by applying the DFS and BFS to multicolored graphs. The case that $a = 1$ was proved earlier in [81, 85] by recurrence relations. In [75] Shin and Zeng constructed a bijective proof for the case that a is a multiple of b, which gives a refinement of Theorem 1.37 in that case.

b-divisible noncrossing partitions

Theorems 1.20 and 1.22 have been generalized by Stanley [79] to the lattice of b-divisible noncrossing partitions. Fix an positive integer b. A *b-divisible noncrossing partition* is a noncrossing partition π for which every block size is divisible by b. Thus π is a noncrossing partition for a set $[bn]$ for some $n \geq 0$. Let $NC_n^{(b)}$ be the poset of all b-divisible noncrossing partitions of $[bn]$. The combinatorial properties of the poset $NC_n^{(b)}$ were first studied by Edelman [26]. In particular, if a pair (π, π') is an edge of $NC_n^{(b)}$, then (π, π') is an edge of NC_{bn}. Hence the edge labeling Λ and Λ_1 of NC_n defined in Section 1.3.2 restrict to edge-labelings of $NC_n^{(b)}$.

Theorem 1.20 is generalized as follows.

Theorem 1.39 *The label $\Lambda_1(\mathfrak{m})$ of the maximal chains of $\mathrm{NC}_{n+1}^{(b)}$ consists of the (b,b)-parking functions of length n, each occurring once.*

The posets $NC_n^{(b)}$ do not have a $\hat{0}$ when $b > 1$. For these posets one regards the minimal elements as having rank 0, and defines $\alpha_{NC_{n+1}^{(b)}}(S)$ and $\beta_{NC_{n+1}^{(b)}}(S)$ for $S \subseteq [n-1]_0$. Theorem 1.22 is generalized as follows.

Theorem 1.40 *Let $S \subseteq [n-1]$.*

1. *The number of* $[b,b]$*-parking functions* $\mathbf{a}$ *of length n with* $\mathrm{Des}(\mathbf{a}) = S$ *is equal to*

$$\beta_{\mathrm{NC}^{(b)}_{n+1}}([n-1]-S)+\beta_{\mathrm{NC}^{(b)}_{n+1}}([n-1]_0-S).$$

2. *The number of* $[b,b]$*-parking functions* $\mathbf{a}$ *of length n satisfying* $S \subseteq \mathrm{Des}(\mathbf{a})$ *is equal to*

$$\alpha_{\mathrm{NC}^{(b)}_{n+1}}([n-1]_0-S).$$

Extended Shi arrangements

For $b \geq 1$ the *extended Shi arrangement* $\mathscr{S}_n^b$ is the collection of hyperplanes

$$x_i - x_j = -b+1, -b+2, \dots, b, \text{ for } 1 \leq i < j \leq n.$$

The labeling of the regions of the Shi arrangement defined in Section 1.3.3 can be extended to this case, as described by Stanley [81].

Define the base region R_0 of $\mathscr{S}_n^b$ by

$$R_0:\ x_1 > x_2 > \cdots > x_n > x_1 - 1.$$

First label the region R_0 by $(0,0,\dots,0) \in \mathbb{N}^n$. Suppose now that R has been labeled, and that R' is an unlabeled region which is separated from R by a unique hyperplane $H: x_i - x_j = m$, where $i < j$. Then define

$$\lambda(R') = \begin{cases} \lambda(R)+e_i, & \text{If } H \text{ is given by } x_i - x_j = m \text{ with } i<j \text{ and } m>0 \\ \lambda(R)+e_j, & \text{if } H \text{ is given by } x_i - x_j = m \text{ with } i<j \text{ and } m \leq 0. \end{cases}$$

Then

Theorem 1.41 *The labels* $\lambda(R)$ *of the extended Shi arrangement* $\mathscr{S}_n^{(b)}$ *are just the* $(1,b)$*-parking functions of length n, each occurring exact once. In addition, the distance enumerator of the extended Shi arrangement is equal to the sum enumerator of* $(1,b)$*-parking functions.*

1.5 Parking Functions Associated with Graphs

1.5.1 *G*-parking functions

In 2004 Postnikov and Shapiro [68] proposed a new generalization of parking functions, namely, the G-parking functions associated with a general connected digraph G. Let G be a directed graph with the vertex set $[n]_0$, where multiple edges and loops are allowed. We will view the vertex 0 as the *root*. As usual in graph theory, a directed edge is represented by a pair (i,j) of vertices, where i is the tail of the edge and j is the head of the edge. For a vertex i, the *indegree* $\text{indeg}(i)$ is the number of edges with tail i, and the *outdegree* $\text{outdeg}(i)$ is the number of edges with head i. In addition, for any subset $U \subseteq [n]$ and vertex $i \in U$, we define $\text{outdeg}_U(i)$ to be the cardinality of the set $\{(i, j) \in E(G) \mid j \notin U\}$, where $E(G)$ is the set of edges of G.

Definition 1.42 *A G-parking function is a function f from $[n]$ to the set of non-negative integers $\mathbb{N}$ satisfying the following condition: for each subset $U \subseteq [n]$ of vertices of G, there exists a vertex $i \in U$ such that $f(i) < \text{outdeg}_U(i)$.*

For the complete graph K_{n+1} on $[n]_0$, such defined functions are exactly the ordinary parking functions if we view K_{n+1} as the digraph with one directed edge (i, j) for each $i \neq j$, and record the function f as a sequence $(f(1), f(2), \ldots, f(n))$.

In a digraph G an *oriented spanning tree* T of G is a subgraph T in which (1) the root 0 has outdegree 0, (2) all other vertices have outdegree 1, and (3) there exists a unique directed path from any vertex i to the root 0. The number of oriented spanning trees of G, denoted $\kappa(G)$, is sometimes called the *complexity* of G. The value of $\kappa(G)$ can be computed by the Matrix-Tree Theorem; see [78, Chap 5.6].

The first important result of G-parking functions is the following theorem that extends the relation between ordinary parking functions and labeled trees.

Theorem 1.43 *The number of G-parking functions equals the number of oriented spanning trees of the digraph G.*

The motivation for Postnikov and Shapiro to work with G-parking functions is to study a general class of algebras formed by taking the quotient of the polynomial ring modulo monotone monomial ideals and their deformations. As a special example, for any digraph G they define two algebras $\mathscr{A}_G$ and $\mathscr{B}_G$ and describe their monomial bases. The basis elements correspond to G-parking functions. Then they proved Theorem 1.43 by applying general formulas for the Hilbert series and dimensions of the algebras given by a monotone monomial ideal.

Shortly after [68] appeared, Chebikin and Pylyavskyy [21] constructed a family of bijections between the set of G-parking functions and the set of oriented spanning trees of the digraph G. Their bijections were built on "different total orders" on the vertices of subtrees of G, similar to the "selection procedure" of Françon's [31] in constructing bijections between ordinary parking functions and rooted labeled trees. We will describe these bijections in a slightly more general setting in Section 1.5.3.

1.5.2 Abelian sandpile model

There is an interesting relation between G-parking functions and the *abelian sandpile model*, one of the archetype models of self-organized criticality in physical systems. The sandpile model was originated from an automaton model by Bak, Tang, and Wiesenfeld [7] for evolutions of dynamical systems on $\mathbb{Z}^2$, and was first formulated by Dhar [25] on any finite connected graphs. The model was also considered by combinatorialists as a game on a graph called the *chip firing game* or the *dollar game*, e.g. [12, 14, 15].

To show its connection to Tutte polynomials, we describe a simple version of the sandpile model for undirected graphs with possible multiple edges but no loops. A review of the more general theory with a class of toppling matrices, as well as the history and references on the subject can be found in the appendix of [68].

Let G be a connected graph on the vertex set $[n]_0$, where the vertex 0 is the root. A configuration (sandpile) is a collection of indistinguishable chips distributed among the vertices of G. More precisely, it is a function u from $[n]$ to $\mathbb{N}$ indicating how many chips are at each vertex $i \neq 0$. A non-root vertex i is called *unstable* if it has at least as many chips as its degree, and an unstable vertex can *topple* by sending chips to adjacent vertices, one along each incident edge. In formula, toppling at vertex i changes a configuration u to u' by

$$u'(j) = \begin{cases} u(i) - \deg(i) & \text{if } j = i, \\ u(j) + e(i,j) & \text{otherwise,} \end{cases}$$

for $j \neq 0$, where $e(i,j)$ is the number of edges between i and j.

In a configuration toppling one vertex may cause neighboring vertices to become unstable. A configuration is said to be *stable* if all the non-root vertices are stable. It is easy to see that for a connected graph G any initial configuration can be transformed into a stable configuration by a sequence of topplings at non-root vertices. Moreover, the final stable configuration does not dependent on the order the topplings, leading to the name "abelian sandpile model".

The root 0 plays a special role, for which we set that $u(0) = -\sum_{i \neq 0} u(i)$. For a stable configuration u, we can let the root topple by increasing the chips at any neighbor i of 0 by $e(0,i)$, where $e(0,i)$ is the number of edges connecting 0 to i, and deceasing the "chips" at the root by $\deg(0)$. By allowing the root to topple for stable configurations, the sandpile model can continue indefinitely, and produce an infinite number of stable configurations. A configuration is *recurrent* in an evolving system if it could be observed after a long period of the evolution of the system. For abelian sandpile model, the system is considered to evolve by adding some chips at a random vertex and then applying the toppling rules. This leads to the following definition.

Definition 1.44 *A configuration u is* recurrent *if it is stable and there exists a positive configuration $v \neq 0$ such that u can be obtained by a sequence of topplings from $u + v$.*

Recurrent configurations are also called *critical configurations* in [12]. There are many characterizations of recurrent configurations, see, for example, [25, 12, 24, 22]. The following explicit characterization is due to Dhar. Let say that a configuration u is *allowed* if for any non-empty subset $I \subseteq [n]$ there exists $j \in I$ such that

$$u(j) \geq \sum_{i \in I \setminus \{j\}} e(i,j).$$

Theorem 1.45 *A configuration is recurrent if and only if it is stable and allowed.*

Comparing Theorem 1.45 and the definition of G-parking functions, we see that recurrent configurations and G-parking functions are complement to each other, as pointed out in [68]. The special case for complete graph K_{n+1} was proved earlier by Cori and Rossin [24].

Theorem 1.46 *Let G be a connected graph on $[n]_0$. A configuration $u : [n] \to \mathbb{N}$ is recurrent if and only only if the function $u^\vee$ given by $u^\vee(i) = \deg(i) - 1 - u(i)$ is a G-parking function.*

Dhar described a simple recursive procedure, the *burning algorithm*, to determine if a given stable configuration is allowed. Translating to G-parking functions, it leads to a linear algorithm that tests whether a given function $u : [n] \to \mathbb{N}$ is a G-parking function: First we mark the root 0. At each iteration of the algorithm, we mark all vertices i that have more marked neighbors than the value $u(i)$. Then u is a G-parking function if and only if all the vertices are marked in the end.

The following example shows how the algorithm works. In Figure 1.8 there are two vertex-functions on the complete graph K_5, where each vertex $i \neq 0$ is labeled by $i/f(i)$, i.e., the first label is the index of the vertex, and the second label is the function value. The first one shows a G-parking function, as one can mark all the vertices in the order $0,1,4,2,3$, while the second is not: once one marks vertices $0,1,4$, no more vertices can be marked.

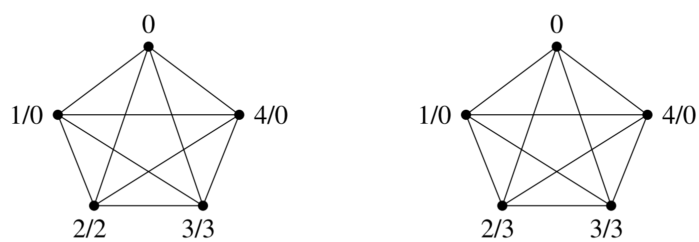

Figure 1.8
Dhar's burning algorithm.

The set of recurrent configurations of a sandpile model on G forms an abelian group whose order is equal to the number of spanning trees of the graph. In particular, let the *level* of a recurrent configuration u be given by

$$\text{level}(u) = \sum_{i \in [n]} u_i + \deg(0) - |E(G)|.$$

Written in terms of G-parking function, the statistic $\text{level}(u)$ induces a *weight* of $u^\vee$ as

$$\text{wt}(u^\vee) = |E(G)| - n - \sum_{i=1}^{n} u^\vee(i). \tag{1.28}$$

Let $P(G,y)$ be the level enumerator of the recurrent configurations of the sandpile model of G, or equivalently, the weight enumerator of G-parking functions, in formula,

$$P(G,y) = \sum_{u} y^{\text{level}(u)} = \sum_{u^\vee} y^{\text{wt}(u^\vee)},$$

where u ranges over all recurrent configurations. The polynomial $P(G,y)$ is the natural generalization of the displacement enumerator of the ordinary parking functions, and is a specialization of the Tutte polynomial $T_G(x,y)$, which we recall next.

Suppose we are given a connected graph G and a total ordering of its edges. Consider a spanning tree T of G. An edge $e \in G - T$ is *externally active* if it is the smallest edge in the unique cycle contained in $T \cup e$. An edge $e \in T$ is *internally active* if it is the smallest edges in the unique cocycle contained in $(G-T) \cup \{e\}$ (a cocycle is a minimal edge cut). We let $\text{ea}(T)$ be the number of externally active edges of T, and $\text{ia}(T)$ the number of internally active edges in T. Tutte [83] then defined his polynomial as

$$T_G(x,y) = \sum_{T \subseteq G} x^{\text{ia}(T)} y^{\text{ea}(T)},$$

where the sum is over all spanning trees T of G. Tutte showed that $T_G(x,y)$ is well-defined, i.e., independent of the total ordering of the edges of G.

The next theorem reveals the relation between $P(G,y)$ and the Tutte polynomial $T_G(x,y)$.

Theorem 1.47 *For a connected graph G,*

$$P(G,y) = T_G(1,y)$$

where $T_G(x,y)$ is the Tutte polynomial of G. In particular, there are as many G-parking functions of weight i as spanning trees of G with external activity i.

Theorem 1.47 was conjectured by Biggs and proved by Merino López [64] by using a recursive characterization of Tutte polynomials. Cori and Le Borgne [23] constructed a bijection between recurrent configurations and spanning trees of G, which carries external activities of the spanning trees to the levels of the recurrent configurations of the sandpile model.

1.5.3 Multiparking functions, graph searching, and the Tutte polynomial

To get a complete expression of the bivariate Tutte polynomial, Kostic and Yan [54] introduced the notion of G-multiparking functions, which are in one-to-one correspondence with spanning forests of the graph G. They described a family of bijections between the spanning forests of a graph G and the G-multiparking functions, each of which corresponds to a graph searching algorithm. In particular, the bijection induced by the breadth-first search leads to a new characterization of external activity, and hence a representation of the bivariate Tutte polynomial by statistics of multiparking functions. This generalizes Theorems 1.5 and 1.47, as well as the correspondence between ordinary parking functions and labeled trees.

For simplicity and clarity, in the following we assume that G is a simple connected graph on $[n]$. The treatment of general directed graphs can be found in [54].

Definition 1.48 *Let G be a simple graph with $V(G) = [n]$. A* G-multiparking function *is a function $f : V(G) = [n] \to \mathbb{N} \cup \{\infty\}$, such that for every $U \subseteq V(G)$ either* **(A)** *i is the vertex of smallest index in U and $f(i) = \infty$, or* **(B)** *there exists a vertex $i \in U$ such that* $0 \le f(v_i) < \mathrm{outdeg}_U(i)$.

The vertices which satisfy $f(i) = \infty$ in **(A)** will be called *roots of f* and those that satisfy **(B)** (in U) are said to be *well-behaved* in U. The G-multiparking functions with only one root (which is necessarily vertex 1) are equivalent to G-parking functions defined by Postnikov and Shapiro.

An easy way to check whether a function is a G-multiparking function is to modify Dhar's burning algorithm to allow the root vertices.

Proposition 1.49 *A vertex function is a G-multiparking function if and only if there exists an ordering $\pi(1), \pi(2), \ldots, \pi(n)$ of the vertices of the graph G such that for every j, $\pi(j)$ satisfies either condition* **(A)** *or condition* **(B)** *in $U_j := \{\pi(j), \ldots, \pi(n)\}$.*

Let $\mathscr{MP}_G$ be the set of G-multiparking functions and $\mathscr{F}_G$ the set of spanning forests of G.

Theorem 1.50 *The set $\mathscr{MP}_G$ are in one-to-one correspondence with the set $\mathscr{F}_G$.*

Again there are many ways to construct bijections from $\mathscr{MP}_G$ to $\mathscr{F}_G$. Extending the idea of Section 1.2.3, we introduce a family of such bijections, each determined by a choice function and corresponds to a searching algorithm on trees/forests.

For a graph G, a *sub-forest F* of G is a subgraph of G without cycles. (Here we don't assume any root in F). A leaf of F is a vertex $v \in V(F)$ of degree 1 in F. Denote the set of leaves of F by Leaf(F). Let $\prod$ be the set of all ordered pairs (F, W) such that F is a sub-forest of G, and $\emptyset \neq W \subseteq$ Leaf(F). As in Section 1.2.3 define a *choice function* γ as a function from $\prod$ to $V(G)$ such that $\gamma(F, W) \in W$.

It is easier to describe the bijection from $\mathscr{F}_G$ to $\mathscr{MP}_G$, hence we present it first in the following Algorithm A. Fix a choice function γ. Let G be a graph on $[n]$ with a spanning forest F. Let $T_1, \ldots, T_k$ be the trees of F with respective minimal vertices $r_1 = 1 < r_2 < \cdots < r_k$.

Algorithm A.

- **Step 1. Determine a processing order π.**
  Define a permutation $\pi = (\pi(1), \pi(2), \ldots, \pi(n)) = (v_1 v_2 \ldots v_n)$ on the vertices of G as follows. First, $v_1 = 1$. Assuming $v_1, v_2, \ldots, v_i$ are determined,
  - If there is no edge of F connecting vertices in $V_i = \{v_1, v_2, \ldots, v_i\}$ to vertices outside V_i, let v_{i+1} be the vertex of smallest index not already in V_i;
  - Otherwise, let $W = \{v \notin V_i : v$ is adjacent to some vertices in $V_i\}$, and F' be the forest obtained by restricting F to $V_i \cup W$. Let $v_{i+1} = \gamma(F', W)$.

- **Step 2. Define a G-multiparking function $f = f_F$.**
  Set $f(r_1) = f(r_2) = \cdots = f(r_k) = \infty$. For any other vertex v, let r_v be the minimal vertex in the tree containing v, and $v, v^p, u_1, \ldots, u_t, r_v$ be the unique path from v to r_v. (That, is, $v^p = \text{pre}(v)$.) Set $f(v)$ to be the cardinality of the set
  $$\{v_j | \{v, v_j\} \in E(G),\ \pi^{-1}(v_j) < \pi^{-1}(v^p)\}.$$

The inverse bijection is quite complicated, which is achieved by the following algorithm B. Given a G-multiparking function $f \in \mathscr{M}\mathscr{P}_G$, Algorithm B finds a spanning forest $F \in \mathscr{F}_G$. Explicitly, we define quadruples $(\text{val}_i, P_i, Q_i, F_i)$ recursively for $i = 0, 1, \ldots, n$, where $\text{val}_i : V(G) \to \mathbb{Z}$ is the *value function*, P_i is the set of *processed* vertices, Q_i is the set of vertices *to be processed*, and F_i is a sub-forest of G with $V(F_i) = P_i \cup Q_i$, $Q_i \subseteq \text{Leaf}(\text{F}_\text{i})$ or Q_i consists of an isolated vertex of F_i.

Algorithm B.

- **Step 1: initial condition.** Let $\text{val}_0 = f$, P_0 be empty, and $F_0 = Q_0 = \{1\}$.
- **Step 2: choose a new vertex v.** At time $i \geq 1$, let $v = \gamma(F_{i-1}, Q_{i-1})$, where γ is the choice function.
- **Step 3: process vertex v.** For every vertex w adjacent to v and $w \notin P_{i-1}$, set $\text{val}_i(w) = \text{val}_{i-1}(w) - 1$. For any other vertex u, set $\text{val}_i(u) = \text{val}_{i-1}(u)$. Let $N = \{w | \text{val}_i(w) = -1, \text{val}_{i-1}(w) \neq -1\}$. Update P_i, Q_i and F_i by letting
  1. $P_i = P_{i-1} \cup \{v\}$,
  2. $Q_i = Q_{i-1} \cup N \setminus \{v\}$ if $Q_{i-1} \cup N \setminus \{v\} \neq \emptyset$; otherwise $Q_i = \{u\}$ where u is the vertex of the lowest-index in $[n] - P_i$.
  3. Let F_i be a graph on $P_i \cup Q_i$ whose edges are obtained from those of F_{i-1} by joining edges $\{w, v\}$ for each $w \in N$.

  We say that the vertex v is processed at time i.

Iterate steps 2-3 until $i = n$. We must have $P_n = [n]$ and $Q_n = \emptyset$. Define $\Phi = \Phi_{\gamma,G} : \mathscr{M}\mathscr{P}_G \to \mathscr{F}_G$ by letting $\Phi(f) = F_n$.

Note that the forest $F = \Phi(f)$ is built tree by tree by Algorithm B. That is, if T_i and T_j are tree components of F with roots r_i, r_j and $r_i < r_j$, then any vertex of tree T_i is processed before any vertex of T_j. The bijection Φ maps G-multiparking functions with k roots to spanning forests with k components.

We explain Algorithms A & B with an example where the choice function is given by the breadth-first order with a queue structure, i.e., the order $<_{bfq}$ described in Example 1.15, with additional rules comparing vertices in different trees. Explicitly, we view the minimal vertex of each tree component as the root of the tree. The order $<_{bfq}$ is defined as follows.

1. Any vertex of tree T_i is less than any vertex of tree T_j if the root of T_i is less than the root of T_j
2. For vertices i, j in the same tree component, let $i <_{bfq} j$ if
   (a) $\text{level}(i) < \text{level}(j)$, or
   (b) $\text{level}(i) = \text{level}(j)$ and $\text{pre}(i) <_{bfq} \text{pre}(j)$, or
   (c) $\text{pre}(i) = \text{pre}(j)$ and $i < j$.

Example 1.51 *A graph G and a multiparking function f are given in Figure 1.9, in which each vertex is labeled by $i/f(i)$, i.e., the first label is the index of the vertex, and the second label is the function value.*

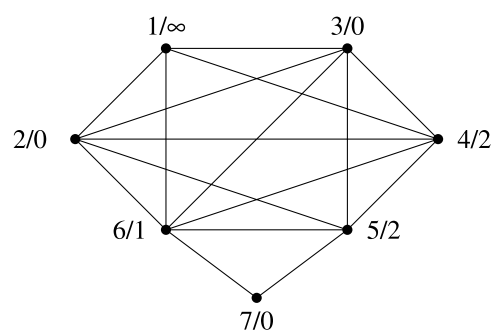

Figure 1.9
A graph and a multiparking function.

The spanning forest created by Algorithm B is indicated by the thick lines in Figure 1.10, while other edges are in dotted lines. In Figure 1.10, each vertex is labeled by $i/\text{val}_n(i)$. The (P_i, Q_i, F_i) for this instance are as follows, where each Q_i is an ordered set, and the first element in Q_i is the next one to be processed. Since P_i and F_i are increasing sets, we just indicate when a new element is added to them.

t	0	1	2	3	4	5	6	7
Q_t	(1)	(2,3)	(3,6)	(6,4)	(4,5,7)	(5,7)	(7)	$\emptyset$
new i in P_t		1	2	3	6	4	5	7
new edges in F_t		{1,2}, {1,3}	{2,6}	{3,4}	{6,5}, {6,7}			

Conversely, starting with a spanning tree, we can easily recover the G-multiparking function using Algorithm A. For the spanning tree in Figure 1.10, the

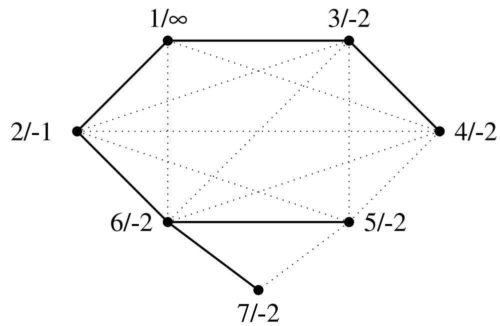

Figure 1.10
A spanning forest found by Algorithm A.

processing order is $1,2,3,6,4,5,7$. We have that $f(1) = \infty$ since it is the minimal element. Let v be the vertex 5. Then $v^p = 6$ and v is connected to two other vertices, namely, vertex 2 and 3, which are less than 6 in the processing order. Hence $f(5) = 2$. Similarly we can compute $f(i)$ for other vertices, and get

$$f(1) = \infty,\ f(2) = f(3) = f(7) = 0,\ f(4) = f(5) = 2,\ f(6) = 1.$$

To get a connection to the Tutte polynomial of G, we use another expression of $T_G(x,y)$. Let H be a (spanning) subgraph of G. Denote by $c(H)$ the number of components of H. Define two invariants associated with H as

$$\sigma(H) = c(H) - 1, \qquad \sigma^*(H) = |E(H)| - |V(G)| + c(H). \tag{1.29}$$

The following alternative description of Tutte polynomials is well-known, for example, see [11].

Theorem 1.52

$$t_G(1+x, 1+y) = \sum_{H \subseteq G} x^{\sigma(H)} y^{\sigma^*(H)}, \tag{1.30}$$

where the sum is over all spanning subgraphs H of G.

For a subgraph H of G, we apply the breadth-first search by using the implementation with a queue structure, as described in Section 1.2.3, to find a spanning forest. Since H may not be connected, we modify the algorithm by adding the least unvisited vertex to the queue whenever the queue is empty but not all the vertices are processed. By the same argument as before, we partition all the subgraphs of G into intervals, one for each spanning forest F. Each interval turns out to be a Boolean algebra consisting of all the ways to add some extra edges to its representing spanning forest. Expressing the Tutte polynomial in terms of sums over such intervals leads to an equation between the Tutte polynomial and a bivariate generating function of G-multiparking functions.

Given a G-multiparking function f, let $\pi = \pi_1\pi_2\ldots\pi_n$ be an ordering of the vertex set $[n]$ determined by applying the burning algorithm in a greedy way: Let $\pi_1 = 1$. After determining $\pi_1,\ldots,\pi_{i-1}$, if $V_i = V(G) - \{\pi_1,\ldots,\pi_{i-1}\}$ has a well-behaved vertex, that is, a vertex $v \in V_i$ such that $0 \le f(v) < \text{outdeg}_{V_i}(v)$, let π_i be one of them; otherwise, let π_i be the minimal vertex of V_i. In the latter case π_i has to be a root of f.

Define the *total root record* $\text{Rec}(f)$ of f as the cardinality of the set

$$\{\{v,w\} \in E(G) \mid \ v \text{ is a root of } f \text{ and } w \text{ appears in front of } v \text{ in the order } \pi\ \}.$$

It can be proved that the value $\text{Rec}(f)$ is independent of the order π and hence well-defined. Let the weight of a G-multiparking function f be

$$\text{wt}(f) = |E(G)| + r(f) - n - \text{Rec}(f) - \big(\sum_{v\neq 0} f(v)\big),$$

where $r(f)$ is the number of roots of f. Then we have

Theorem 1.53 *For any connected graph G,*

$$T_G(1+x,y) = \sum_{f\in\mathscr{M}\mathscr{P}(G)} x^{r(f)-1}y^{\text{wt}(f)}.$$

where $\mathscr{M}\mathscr{P}(G)$ is the set of G-multiparking functions on G.

Note that if $r(f) = 1$, the function f can only have one root at vertex 1, therefore $\text{Rec}(f) = 0$ and $\text{wt}(f) = |E(G)| - n + 1 - \big(\sum_{v\neq 1} f(v)\big)$. Hence Theorem 1.47 is a special case of Theorem 1.53 by taking $x = 0$.

Corollary 1.54 *For connected graph G, the number of G-multiparking functions is $t_G(2,1)$. Among them, those with an odd number of roots is counted by $\frac{1}{2}(t_G(2,1)+t_G(0,1))$, and those with an even number of roots is counted by $\frac{1}{2}(t_G(2,1)-t_G(0,1))$.*

Proof. The first sentence is obtained by taking $x = y = 1$ in Theorem 1.53. For the second sentence, one notices that $t_G(0,1) = \sum_f (-1)^{r(f)-1}$ is the difference between the number of G-multiparking functions with an odd number of roots, and those with an even number of roots. □

Theorem 1.53 also leads to a new expression of $t_{K_{n+1}}(x,y)$, the Tutte polynomial of complete graphs with $n+1$ vertices, in terms of ordinary parking functions. The exponent of the variable x enumerates ordinary parking functions by the number of *critical left-to-right maxima*. Given an ordinary parking function $\mathbf{b} = (b_1,\ldots,b_n)$, we say that a term $b_i = j$ is *critical* if in $\mathbf{b}$ there are exactly j terms less than j, and exactly $n-1-j$ terms larger than j. For example, in $\mathbf{b} = (3,0,0,2)$, the terms $b_1 = 3$ and $b_4 = 2$ are critical. Among them, only $b_1 = 3$ is also a left-to-right maximum.

Let $\alpha(\mathbf{a})$ be the number of critical left-to-right maxima in an ordinary parking function $\mathbf{a}$. We have

Theorem 1.55
$$t_{K_{n+1}}(x,y) = \sum_{\mathbf{a}\in\mathscr{P}\mathscr{K}_n} x^{\alpha(\mathbf{a})} y^{\binom{n}{2}-\sum_i b_i},$$
where $\mathscr{P}\mathscr{K}_n$ is the set of ordinary parking functions of length n.

An interesting question is how to express the bivariate Tutte polynomials in terms of G-parking functions only, avoiding introducing roots or using spanning forests. An answer was provided by Chang, Ma and Yeh [19], who proposed a notion of *critical-bridge* for G-parking functions, and proved that

$$T_G(x,y) = \sum_f x^{\mathrm{cb}(f)} y^{\mathrm{wt}(f)},$$

where $\mathrm{cb}(f)$ is the number of critical bridges of f, and the sum ranges over all G-parking functions.

1.6 Final Remarks

We close this survey by briefly mentioning a fast growing area, the combinatorial study of Macdonald polynomials, in which parking functions play an important role. Macdonald polynomials [63] is a family of multivariable orthogonal polynomials with applications to a wide variety of subjects, including algebraic combinatorics, harmonic analysis, Hilbert schemes, and representation theory. The combinatorics theory behind Macdonald polynomials was pioneered by Garsia, Haiman, and their school, and was developed dramatically in the last 20 years, with exciting new problems and progressions. Parking functions and their encoding as labeled Dyck paths provide powerful tools to study the intrinsic combinatorial structures. Conversely, the theory of Macdonald polynomials leads to many new generalizations and extensions that greatly enrich the theory of parking functions.

A comprehensive coverage on the combinatorics of Macdonald polynomials and the space of diagonal harmonics is given in the monograph [43] by Haglund. The enumerative results associated to statistics on parking functions and labeled lattice paths, as well as the two-parameter versions of related objects such as Catalan numbers and Schröder paths, are included in the chapter *Catalan Paths and q,t-enumeration* of this book. Here we give a gentle introduction of the role played by parking functions and the connection to previous sections.

Parking functions appear in the spaces of diagonal harmonics and diagonal coinvariants, which concern the action of the symmetric group $\mathfrak{S}_n$ on two sets $(x_1,\dots,x_n)$ and $(y_1,\dots,y_n)$ of n variables. For a polynomial in $R_n = \mathbb{C}[x_1,\dots,x_n,y_1,\dots,y_n]$, let $\mathfrak{S}_n$ act diagonally, i.e.,

$$w\cdot x_i = x_{w(i)}, \qquad w\cdot y_i = y_{w(i)}.$$

It is known that

$$\{\sum_{i=1}^{n} x_i^h y_i^k, h+k \geq 0\} \tag{1.31}$$

generate $R_n^{\mathfrak{S}_n}$, the ring of invariants under the diagonal action. The *quotient ring DR_n of diagonal coinvariants* is defined by $DR_n = R_n/I$, where I is the ideal of R_n generated by elements of (1.31) with zero constant term. By analogy the *space of diagonal harmonics DH_n* is defined by

$$DH_n = \left\{ f \in R_n \ : \ \sum_{i=1}^{n} \frac{\partial^h}{x_i^h} \frac{\partial^k}{y_j^k} f = 0, \forall h+k > 0 \right\}.$$

The spaces DR_n and DH_n are finite dimensional isomorphic vector spaces. A major theorem of DH_n is the so-called $(n+1)^{n-1}$ conjecture, which was proposed by Garsia and Haiman [34] and later proved by Haiman [46].

Theorem 1.56 *The dimension of DH_n is $(n+1)^{n-1}$.*

Let $V_{h,k,n}$ be he set of polynomials $f \in R_n$ such that f is homogeneous of degree h in the x-variables and homogeneous of degree k in the y-variables. By convention, the zero polynomial belongs to every $V_{h,k,n}$. We have the decomposition

$$R_n = \bigoplus_{h \geq 0} \bigoplus_{k \geq 0} V_{h,k,n},$$

which turns R_n into a doubly graded vector space. One can write

$$DH_n = \bigoplus_{h \geq 0} \bigoplus_{k \geq 0} (DH_n \cap V_{h,k,n}).$$

The *Hilbert series of diagonal harmonics* is the bivariate polynomial

$$H_n(q,t) = \sum_{h \geq 0} \sum_{k \geq 0} \dim(DH_n \cap V_{h,k,n}) q^h t^k.$$

Similarly, one can turn DR_n into a doubly graded vector space. Ignoring the ring structure of DR_n, the spaces DR_n and DH_n are isomorphic as doubly graded $\mathfrak{S}_n$-modules.

Haiman [46] proved that the Hilbert series $H_n(q,t)$ has the following properties.

1. $H_n(1,1) = (n+1)^{n-1}$.
2. $q^{n(n-1)/2} H_n(1/q,q) = (1+q+\cdots+q^n)^{n-1}$.
3. $H_n(q,t) = H_n(t,q)$.

By the first property, $H_n(q,t)$ is a sum of $(n+1)^{n-1}$ monomials $q^h t^k$. This suggests that there could be a combinatorial interpretation of $H_n(q,t)$ as a joint enumerator of two statistics over parking functions. More precisely, one seeks two functions stat_1 and stat_2 from $\mathscr{PK}_n$ to $\mathbb{N}$ such that

$$H_n(q,t) = \sum_{\alpha \in \mathscr{PK}_n} q^{\text{stat}_1(\mathbf{a})} t^{\text{stat}_2(\mathbf{a})}.$$

The first conjectured combinatorial formula for $H_n(q,t)$ was proposed by Haglund and Loehr [45], and involved statistics "area" and "dinv". These two statistics are defined on labeled Dyck paths, a geometric encoding of parking functions.

A *Dyck path of order n* is a lattice path from $(0,0)$ to (n,n) consisting of steps East ($E = (1,0)$) and North ($N = (0,1)$), and never going below the line $y = x$. A *labeled Dyck path* is a Dyck path in which the n north steps are labeled 1 to n in such a way that the labels of consecutive north steps increase from bottom to top. See Figure 1.11 for an example, where one places each label in the lattice square to the right of the corresponding north step.

There is a simple bijection between labeled Dyck paths and parking functions. Let P be a labeled Dyck path. For each $i \in [n]$, find the north step with label i and let a_i be the x-coordinate of this step. The fact that P is a Dyck path implies that the sequence $(a_1, \ldots, a_n)$ is a parking function. Conversely, given a parking function, one places all cars that prefer the space j in column $j+1$ in increasing order, such that there is exactly one car in each row, and cars preferring earlier spots appear in lower rows of the figure. The labels determine a labeled lattice path in the obvious way, which is easily seen to be a Dyck path. Example 1.57 shows this correspondence.

Example 1.57 *For the parking function* $\mathbf{a} = (2,0,1,0,1,3,6,0)$*, Figure 1.11 shows the corresponding labeled Dyck path. To see how the labels are assigned, take the first column as an example. In* $\mathbf{a}$ *there are three* 0*s, which means that cars* C_2*,* C_4 *and* C_8 *prefer space 0. Hence there are three north steps in the Dyck path at* $x = 0$*, with labels* $2,4,8$ *from bottom to top.*

The statistic $\text{area}(\mathbf{a})$ of a parking function $\mathbf{a}$ is equivalent to the total displacement $D(\mathbf{a})$ defined in Section 2. The other statistic, $\text{dinv}(\mathbf{a})$, is obtained by counting certain inversion pairs between rows of the labeled Dyck path. The exact definitions and detailed descriptions of these two statistics are given in the chapter *Catalan Paths and* q,t*-enumeration*, and hence omitted here. Let

$$CH_n(q,t) = \sum_{\mathbf{a} \in \mathscr{PK}_n} q^{\text{dinv}(\mathbf{a})} t^{\text{area}(\mathbf{a})}.$$

Haglund and Loehr [45] conjectured that $CH_n(q,t) = H_n(q,t)$. This conjecture has been verified in Maple for $n \leq 11$. The truth of the conjecture when $q = 1$ follows from results of Garsia and Haiman in [35]. It is also the reason that one considers parking functions but not other combinatorial objects counted by $(n+1)^{n-1}$ here: when $q = 1$ there is a natural action of $\mathfrak{S}_n$ on the set $\mathscr{PK}_n$ which gives the Frobenius characteristic of the diagonal harmonics, up to the sign representation. When $t = 1$, the conjecture follows from a result of Loehr and Remmel [62], who proved bijectively that area and dinv have the same distribution. In addition, Loehr [61] obtained

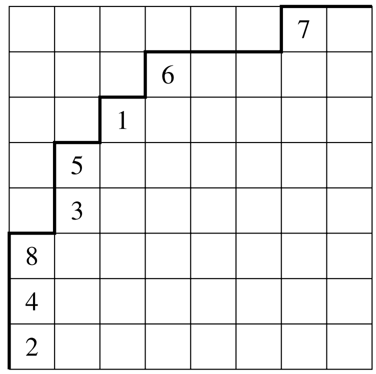

Figure 1.11
The parking function $\mathbf{a} = (2,0,1,0,1,3,6,0)$ and the corresponding labeled Dyck path.

a recursion characterizing the polynomial $CH_n(q,t)$, and proved the specialization

$$q^{n(n-1)/2}CH_n(1/q,q) = (1+q+\cdots+q^n)^{n-1}.$$

The joint symmetry of $CH_n(q,t)$, i.e., $CH_n(q,t) = CH_n(t,q)$, as well as Haglund and Loehr's conjecture are still open. They are part of a larger program on finding the combinatorial description for the character of the space of diagonal harmonics DH_n, which is the same as the character of diagonal coinvariants DR_n. It is known that the character of DR_n as a doubly-graded $\mathfrak{S}_n$-module can be expressed using the Frobenius characteristic map as ∇e_n, where e_n is the n-th elementary symmetric function, and ∇ is an operator from the theory of Macdonald polynomials [46]. In 2005 Haglund, Haiman, Loehr, Remmel, and Ulyanov [44] proposed an explicit combinatorial formula of the expansion of ∇e_n into monomials. It is usually referred to as the "shuffle conjecture" since the formula can be described in terms of statistics on certain permutations associated to parking functions which are shuffles of blocks of increasing and decreasing sequences. The shuffle conjecture contains a Garsia-Haglund formula for the (q,t)-Catalan number $C_n(q,t)$ [32, 33], the Haglund-Loehr conjecture for $H_n(q,t)$, and a formula for (q,t)-Schröder polynomials [42] as special cases.

In the literature on diagonal harmonics, ordinary parking functions enumerated by $(n+1)^{n-1}$ are called "classical parking functions", and it is standard to encode them as labeled Dyck paths. There are several extensions and generalizations of the shuffle conjecture, including an m-parameter extension [44], the *rational shuffle conjecture* [40], and the *compositional rational shuffle conjecture* [10]. The combinatorial foundations of these extensions are given by rational Dyck paths and rational parking functions. Explicitly, let a,b be positive integers. An (a,b)-Dyck paths is a lattice path from $(0,0)$ to (b,a) consisting of N and E steps, and never going below the line $y = \frac{a}{b}x$. A rational (a,b)-parking function is an (a,b)-Dyck path together with a labeling of the north steps by the set $[a]$ such that labels increase in each column from bottom to top. Note that when $b = ka$ for some integer k, the rational (a,b)-parking functions are the same as the classical $(1,k)$-parking functions of length a discussed in Section 1.4.4. For other cases, rational parking functions are not equivalent to **u**-parking functions with an arithmetic **u**. In particular, rational (a,b)-parking functions are counted by b^{a-1} for co-prime positive integers a,b, where the ordinary case corresponds to $(a,b) = (n,n+1)$. The combinatorial theory of rational Dyck paths and rational parking functions, as well as their relations with the representation theory of diagonal coinvariants are discussed in [3].

Acknowledgement

First I would like to express my deepest gratitude to Gian-Carlo Rota and Richard Stanley for introducing me to the wonderful world of combinatorics, in particular, the subject of parking functions, and for their constant supports and encouragements throughout my career. Many colleagues have contributed valuable ideas and comments, by either joint works or personal communications, for which I thank them: Drew Armstrong, Miklós Bóna, William Chen, Ira Gessel, Jim Haglund, Niraj Khare, Dimitrije Kostic, Joseph Kung, Rudolph Lorentz, Svetlana Poznanovik, Bruce Sagan, Joel Spencer, Xinyu Sun, Michelle Wachs, Yeong-Nan Yeh. I am also grateful to Hua Peng for his help on editing the manuscript, and to an anonymous referee for many helpful comments and suggestions.

This paper was written while I was taking a faculty development leave, and was supported by the National Science Foundation and the Association of Former Students at Texas A&M University.

References

[1] D. Armstrong. Generalized noncrossing partitions and combinatorics of Coxeter groups. *Mem. Amer. Math. Soc.*, 202(949), 2009.

[2] D. Armstrong. Hyperplane arrangements and diagonal harmonics. In *23rd International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2011)*, Discrete Math. Theor. Comput. Sci. Proc., AO, pages 39–50. Assoc. Discrete Math. Theor. Comput. Sci., Nancy, 2011.

[3] D. Armstrong, N. Loehr, and G. S. Warrington. Rational parking functions and catalan numbers. 2014. arXiv:1403.1845.

[4] D. Armstrong and B. Rhoades. The Shi arrangement and the Ish arrangement. *Trans. Amer. Math. Soc.*, 364(3):1509–1528, 2012.

[5] C. A. Athanasiadis and S. Linusson. A simple bijection for the regions of the Shi arrangement of hyperplanes. *Discrete Math.*, 204(1-3):27–39, 1999.

[6] M. D. Atkinson and M. Thiyagarajah. The permutational power of a priority queue. *BIT*, 33(1):2–6, 1993.

[7] P. Bak, C. Tang, and K. Wiesenfeld. Self-organized criticality: an explanation of $1/f$ noise. *Phys. Rev. Letters*, 59(4):381–384, 1987.

[8] H. W. Becker. Planar rhyme schemes. *Bull. Amer. Math. Soc.*, 58:39, 1952.

[9] J. S. Beissinger and U. N. Peled. A note on major sequences and external activity in trees. *Electron. J. Combin.*, 4(2):Research Paper 4, 1997.

[10] F. Bergeron, A. Leven, A. Garsia, and G. Xin. Compositional (km,kn)shuffle conjectures. 2014. arXiv:1404.4616.

[11] N. L. Biggs. *Algebraic graph theory*. Cambridge Mathematical Library. Cambridge University Press, Cambridge, second edition, 1993.

[12] N. L. Biggs. Chip-firing and the critical group of a graph. *J. Algebraic Combin.*, 9(1):25–45, 1999.

[13] A. Björner. The homology and shellability of matroids and geometric lattices. In *Matroid applications*, volume 40 of *Encyclopedia Math. Appl.*, pages 226–283. Cambridge Univ. Press, Cambridge, 1992.

[14] A. Björner and L. Lovász. Chip-firing games on directed graphs. *J. Algebraic Combin.*, 1(4):305–328, 1992.

[15] A. Björner, L. Lovász, and P. W. Shor. Chip-firing games on graphs. *European J. Combin.*, 12(4):283–291, 1991.

[16] M. Bousquet-Mélou and G. Schaeffer. Enumeration of planar constellations. *Adv. in Appl. Math.*, 24(4):337–368, 2000.

[17] P. J. Cameron, D. Johannsen, T. Prellberg, and P. Schweitzer. Counting defective parking functions. *Electron. J. Combin.*, 15(1):Research Paper 92, 2008.

[18] A. Cayley. A theorem on trees. *Quart. J. Math.*, 23:376–378, 1889.

[19] H. Chang, J. Ma, and Y.-N. Yeh. Tutte polynomials and G-parking functions. *Adv. in Appl. Math.*, 44(3):231–242, 2010.

[20] P. Chassaing and J.-F. Marckert. Parking functions, empirical processes, and the width of rooted labeled trees. *Electron. J. Combin.*, 8(1):Research Paper 14, 2001.

[21] D. Chebikin and P. Pylyavskyy. A family of bijections between G-parking functions and spanning trees. *J. Combin. Theory Ser. A*, 110(1):31–41, 2005.

[22] R. Cori, A. Dartois, and D. Rossin. Avalanche polynomials of some families of graphs. In *Mathematics and computer science. III*, Trends Math., pages 81–94. Birkhäuser, Basel, 2004.

[23] R. Cori and Y. Le Borgne. The sand-pile model and Tutte polynomials. *Adv. in Appl. Math.*, 30(1-2):44–52, 2003.

[24] R. Cori and D. Rossin. On the sandpile group of dual graphs. *European J. Combin.*, 21(4):447–459, 2000.

[25] D. Dhar. Self-organized critical state of sandpile automaton models. *Phys. Rev. Lett.*, 64(14):1613–1616, 1990.

[26] P. H. Edelman. Chain enumeration and noncrossing partitions. *Discrete Math.*, 31(2):171–180, 1980.

[27] S-P. Eu, T-S. Fu, and C-J. Lai. On the enumeration of parking functions by leading terms. *Adv. in Appl. Math.*, 35(4):392–406, 2005.

[28] S-P. Eu, T-S. Fu, and C-J. Lai. Cycle lemma, parking functions and related multigraphs. *Graphs Combin.*, 26(3):345–360, 2010.

[29] P. Flajolet, P. Poblete, and A. Viola. On the analysis of linear probing hashing. Average-case analysis of algorithms. *Algorithmica*, 22(4):490–515, 1998.

[30] D. Foata and J. Riordan. Mappings of acyclic and parking functions. *Aequationes Math.*, 10:10–22, 1974.

[31] J. Françon. Acyclic and parking functions. *J. Combinatorial Theory Ser. A*, 18:27–35, 1975.

[32] A. M. Garsia and J. Haglund. A positivity result in the theory of Macdonald polynomials. *Proc. Natl. Acad. Sci. USA*, 98(8):4313–4316, 2001.

[33] A. M. Garsia and J. Haglund. A proof of the q,t-Catalan positivity conjecture. *Discrete Math.*, 256(3):677–717, 2002.

[34] A. M. Garsia and M. Haiman. A graded representation model for Macdonald's polynomials. *Proc. Nat. Acad. Sci. U.S.A.*, 90(8):3607–3610, 1993.

[35] A. M. Garsia and M. Haiman. A remarkable q,t-Catalan sequence and q-Lagrange inversion. *J. Algebraic Combin.*, 5(3):191–244, 1996.

[36] I. M. Gessel and B. E. Sagan. The Tutte polynomial of a graph, depth-first search, and simplicial complex partitions. *Electron. J. Combin.*, 3(2):Research Paper 9, 1996.

[37] I. M. Gessel and S. Seo. A refinement of Cayley's formula for trees. *Electron. J. Combin.*, 11(2):Research Paper 27, 2004/06.

[38] I. M. Gessel and D. L. Wang. Depth-first search as a combinatorial correspondence. *J. Combin. Theory Ser. A*, 26(3):308–313, 1979.

[39] J. D. Gilbey and L. H. Kalikow. Parking functions, valet functions and priority queues. *Discrete Math.*, 197/198:351–373, 1999. 16th British Combinatorial Conference (London, 1997).

[40] E Gorsky and A Negut. Refined knot invariants and Hilbert schemes. 2013. arXiv: 1304.3328.

[41] I. P. Goulden and D. M. Jackson. Transitive factorisations into transpositions and holomorphic mappings on the sphere. *Proc. Amer. Math. Soc.*, 125(1):51–60, 1997.

[42] J. Haglund. A proof of the q,t-Schröder conjecture. *Int. Math. Res. Not.*, (11):525–560, 2004.

[43] J. Haglund. *The q,t-Catalan numbers and the space of diagonal harmonics*, volume 41 of *University Lecture Series*. American Mathematical Society, Providence, RI, 2008.

[44] J. Haglund, M. Haiman, N. Loehr, J. B. Remmel, and A. Ulyanov. A combinatorial formula for the character of the diagonal coinvariants. *Duke Math. J.*, 126(2):195–232, 2005.

[45] J. Haglund and N. Loehr. A conjectured combinatorial formula for the Hilbert series for diagonal harmonics. *Discrete Math.*, 298(1-3):189–204, 2005.

[46] M. Haiman. Vanishing theorems and character formulas for the Hilbert scheme of points in the plane. *Invent. Math.*, 149(2):371–407, 2002.

[47] S. Janson. Asymptotic distribution for the cost of linear probing hashing. *Random Structures Algorithms*, 19(3-4):438–471, 2001.

[48] S. Janson, D. E. Knuth, T. Łuczak, and B. Pittel. The birth of the giant component. *Random Structures Algorithms*, 4(3):231–358, 1993.

[49] N. Khare, R. Lorentz, and C. H. Yan. Bivariate gončarov polynomials and integer sequences. *Sci. China Math.*, 57(8):1561–1578, 2014.

[50] D. Kim and S. Seo. Transitive cycle factorizations and prime parking functions. *J. Combin. Theory Ser. A*, 104(1):125–135, 2003.

[51] D. E. Knuth. *The art of computer programming. Volume 3*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1973. Sorting and searching, Addison-Wesley Series in Computer Science and Information Processing.

[52] D. E. Knuth. Linear probing and graphs. *Algorithmica*, 22(4):561–568, 1998.

[53] A. G. Konheim and B. Weiss. An occupancy discipline and applications. *SIAM J. Appl. Math.*, 14:12661274, 1966.

[54] D. Kostić and C. H. Yan. Multiparking functions, graph searching, and the Tutte polynomial. *Adv. in Appl. Math.*, 40(1):73–97, 2008.

[55] G. Kreweras. Sur les partitions non croisées d'un cycle. *Discrete Math.*, 1(4):333–350, 1972.

[56] G. Kreweras. Une famille de polynômes ayant plusieurs propriétés énumeratives. *Period. Math. Hungar.*, 11(4):309–320, 1980.

[57] J. P. S. Kung, X. Sun, and C. H. Yan. Goncarov-type polynomials and applications in combinatorics. *Preprint*, 2007. Available at http://www.math.tamu.edu/ cyan/Files/DGP.pdf.

[58] J. P. S. Kung and C. H. Yan. Exact formulas for moments of sums of classical parking functions. *Adv. in Appl. Math.*, 31(1):215–241, 2003.

[59] J. P. S. Kung and C. H. Yan. Expected sums of general parking functions. *Ann. Comb.*, 7(4):481–493, 2003.

[60] J. P. S. Kung and C. H. Yan. Gončarov polynomials and parking functions. *J. Combin. Theory Ser. A*, 102(1):16–37, 2003.

[61] N. A. Loehr. Combinatorics of q, t-parking functions. *Adv. in Appl. Math.*, 34(2):408–425, 2005.

[62] N. A. Loehr and J. B. Remmel. Conjectured combinatorial models for the Hilbert series of generalized diagonal harmonics modules. *Electron. J. Combin.*, 11(1):Research Paper 68, 2004.

[63] I. G. Macdonald. *Symmetric functions and Hall polynomials*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University Press, New York, second edition, 1995.

[64] C. Merino López. Chip firing and the Tutte polynomial. *Ann. Comb.*, 1(3):253–259, 1997.

[65] P. Moszkowski. Arbres et suites majeures. *Period. Math. Hungar.*, 20(2):147–154, 1989.

[66] A. Nica and R. Speicher. *Lectures on the combinatorics of free probability*, volume 335 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2006.

[67] J. Pitman and R. P. Stanley. A polytope related to empirical distributions, plane trees, parking functions, and the associahedron. *Discrete Comput. Geom.*, 27(4):603–634, 2002.

[68] A. Postnikov and B. Shapiro. Trees, parking functions, syzygies, and deformations of monomial ideals. *Trans. Amer. Math. Soc.*, 356(8):3109–3142, 2004.

[69] Y. Poupard. Étude et dénombrement parallèles des partitions non-croisées d'un cycle et des découpages d'un polygone convexe. *Discrete Math.*, 2(3):279–288, 1972.

[70] H. Prüfer. Neuer beweis eines satzes ber permutationen. *Arch. Math. Phys.*, 27:742–744, 1918.

[71] J. Riordan. Ballots and trees. *J. Combinatorial Theory*, 6:408–411, 1969.

[72] M. P. Schützenberger. On an enumeration problem. *J. Combinatorial Theory*, 4:219–221, 1968.

[73] S. Seo and H. Shin. A generalized enumeration of labeled trees and reverse Prüfer algorithm. *J. Combin. Theory Ser. A*, 114(7):1357–1361, 2007.

[74] J. Y. Shi. *The Kazhdan-Lusztig cells in certain affine Weyl groups*, volume 1179 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1986.

[75] H. Shin and J. Zeng. A further correspondence between $(bc,\bar{b})$-parking functions and $(bc,\bar{b})$-forests. In *21st International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2009)*, Discrete Math. Theor. Comput. Sci. Proc., AK, pages 793–804. Assoc. Discrete Math. Theor. Comput. Sci., Nancy, 2009.

[76] J. Spencer. Enumerating graphs and Brownian motion. *Comm. Pure Appl. Math.*, 50(3):291–294, 1997.

[77] J. Spencer and C. H. Yan. An enumeration problem and branching processes. *Preprint*, 1999. Available at http://www.math.tamu.edu/ cyan/Files/Ball.pdf.

[78] R. P. Stanley. *Enumerative combinatorics. Vol. I*. The Wadsworth & Brooks/Cole Mathematics Series. Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA, 1986.

[79] R. P. Stanley. Hyperplane arrangements, interval orders, and trees. *Proc. Nat. Acad. Sci. U.S.A.*, 93(6):2620–2625, 1996.

[80] R. P. Stanley. Parking functions and noncrossing partitions. *Electron. J. Combin.*, 4(2):Research Paper 20, 1997.

[81] R. P. Stanley. Hyperplane arrangements, parking functions and tree inversions. In *Mathematical essays in honor of Gian-Carlo Rota (Cambridge, MA, 1996)*,

volume 161 of *Progr. Math.*, pages 359–375. Birkhäuser Boston, Boston, MA, 1998.

[82] G. P. Steck. The Smirnov two sample tests as rank tests. *Ann. Math. Statist.*, 40:1449–1466, 1969.

[83] W. T. Tutte. A contribution to the theory of chromatic polynomials. *Canadian J. Math.*, 6:80–91, 1954.

[84] E. M. Wright. The number of connected sparsely edged graphs. *J. Graph Theory*, 1(4):317–330, 1977.

[85] C. H. Yan. Generalized tree inversions and k-parking functions. *J. Combin. Theory Ser. A*, 79(2):268–280, 1997.

[86] C. H. Yan. On the enumeration of generalized parking functions. In *Proceedings of the Thirty-first Southeastern International Conference on Combinatorics, Graph Theory and Computing (Boca Raton, FL, 2000)*, volume 147, pages 201–209, 2000.

[87] C. H. Yan. Generalized parking functions, tree inversions, and multicolored graphs. *Adv. in Appl. Math.*, 27(2-3):641–670, 2001.

[88] C. Zara. Parking functions, stack-sortable permutations, and spaces of paths in the Johnson graph. *Electron. J. Combin.*, 9(2):Research paper 11, 2002/03.

Index