A generalization of parking functions allowing backward movement

Alex Christensen

Department of Mathematics
University of Arizon
Tucson, Arizona, U.S.A.

ajc333@comcast.net

Pamela E. Harris

Department of Mathematics and Statistics
Williams College
Williamstown, MA, U.S.A.

peh2@williams.edu

Zakiya Jones

Department of Mathematics
Pomona College
Claremont, CA, U.S.A.

zakiyacmjones@gmail.com

Marissa Loving

Department of Mathematics
Georgia Institute of Technology
Atlanta, GA, U.S.A.

mloving6@gatech.edu

Andrés Ramos Rodríguez

Department of Computer Science
University of Puerto Rico, Rio Piedras
San Juan, PR, U.S.A.

ramosandres443@gmail.com

Joseph Rennie

Department of Mathematics
University of Illinois at Urbana-Champaign
Champaign, IL, U.S.A.

rennie2@illinois.edu

Gordon Rojas Kirby

Department of Mathematics
University of California, Santa Barbara
Santa Barbara, CA, U.S.A.

gkirby@math.ucsb.edu

Abstract

Classical parking functions are defined as the parking preferences for n cars driving (from west to east) down a one-way street containing parking spaces labeled from 1 to n (from west to east). Cars drive down the street toward their preferred spot and park there if the spot is available. Otherwise, the car continues driving

down the street and takes the first available parking space, if such a space exists. If all cars can park using this parking rule, we call the n-tuple containing the cars' parking preferences a parking function.

In this paper, we introduce a generalization of the parking rule allowing cars whose preferred space is taken to first proceed up to k spaces west of their preferred spot to park before proceeding east if all of those k spaces are occupied. We call parking preferences which allow all cars to park under this new parking rule k-Naples parking functions of length n. This generalization gives a natural interpolation between classical parking functions, the case when $k = 0$, and all n-tuples of positive integers 1 to n, the case when $k \geqslant n - 1$. Our main result provides a recursive formula for counting k-Naples parking functions of length n. We also give a characterization for the $k = 1$ case by introducing a new function that maps 1-Naples parking functions to classical parking functions, i.e. 0-Naples parking functions. Lastly, we present a bijection between k-Naples parking functions of length n whose entries are in weakly decreasing order and a family of signature Dyck paths.

Mathematics Subject Classifications: 05A19

1   Introduction

Parking functions were introduced independently by Ronald Pyke and by Alan Konheim and Benjamin Weiss in relation to hashing problems [5, 6]. Parking functions are combinatorial objects defined as follows. Let the set of natural numbers be defined as $\mathbb{N} := \{1, 2, 3, \dots\}$, and for $n \in \mathbb{N}$ let $[n] := \{1, \dots, n\}$. Now, consider n parking spaces on a one-way street arranged in a line numbered 1 to n from west to east. Suppose there are n cars, denoted $c_1, c_2, \dots, c_n$, that drive in order down this one-way street. For all $1 \leqslant i \leqslant n$, each car c_i has a preferred parking spot $a_i \in [n]$ and multiple cars are allowed to have the same preference. This is illustrated[1] in Figure 1.
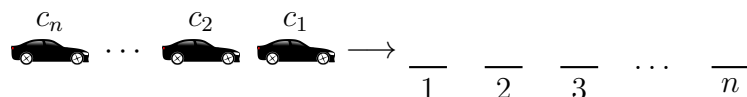

Figure 1: Parking function illustration.

A **parking preference** of length n is an n-tuple of integers in $[n]$ where the i-th component corresponds to the preferred parking spot of car c_i. We denote the set of parking preferences of length n as PP_n. Note that $|PP_n| = n^n$. For a parking preference $\alpha = (a_1, \dots, a_n) \in PP_n$, we establish the following parking rule: for all $1 \leqslant i \leqslant n$, c_i starts at parking space 1 and drives toward its preferred parking spot a_i. If a_i is unoccupied c_i parks. Otherwise, c_i proceeds forward until it reaches the next available parking spot. If every parking spot numbered from a_i up to and including n is taken, then c_i is unable to park. On the other hand, if every car is able to park given the preference $\alpha \in PP_n$, then we say that α is a **parking function**. A necessary and sufficient condition to determine if a parking preference $\alpha = (a_1, a_2, \dots, a_n) \in PP_n$ is a parking function is determined

---

[1]Black car vector. Digital image. The London Telegraph. 13 August 2019, https://www.goodfreephotos.com/vector-images/black-car-vector.png.php.

by considering $\beta = (b_1, \ldots, b_n)$ which is the increasing rearrangement of the entries in α. Then, α is a parking function if and only if $b_i \leqslant i$ for each i. We denote the set of all parking functions of length n as PF_n. It is known that $|PF_n| = (n+1)^{n-1}$ (see [5]).

Parking functions are interesting in their own right and have applications in combinatorics, group theory, the study of hyperplane arrangements, and computer science. Many generalizations of parking functions exist and the main results give formulas to count the number of generalized parking functions. For example, the case where there are more parking spots than cars is considered in [5, Lemma 2] and counted by $(n+1-m)(n+1)^{m-1}$, where m is the number of cars and n is the number of spaces in the lot, with $n \geqslant m$. Another generalization of parking functions given in [8], known as **x**-parking functions, are defined by generalizing the necessary and sufficient condition so that given $\alpha \in PP_n$ and a vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}^n$, α is an **x**-parking function if its increasing rearrangement $\beta = (b_1, \ldots, b_n)$ satisfies $b_i \leqslant x_1 + \cdots + x_i$ for each i. For a survey of classical parking functions and their generalizations, we refer the reader to [9].

In this paper, we study a new generalization of parking functions, introduced by Baumgardner in [1], called **Naples parking functions**. In this generalization, the parking rule for the parking preference $\alpha = (a_1, a_2, \ldots, a_n)$ is as follows. Car c_i drives to its preferred parking spot a_i, and if the spot is empty c_i parks there. Otherwise, c_i first checks back to see if parking spot $a_i - 1$ (the one directly behind its preferred parking spot) is available. If spot $a_i - 1$ is empty and $a_i - 1 \geqslant 1$, c_i parks there. Otherwise, c_i continues east and parks in the first unoccupied spot. If under this new parking rule the parking preference α allows all cars to park, then we call α a Naples parking function. We extend this parking rule by allowing a car that finds its preferred parking spot occupied to look back up to k spaces, for $0 \leqslant k < n$. The car backs up one space at a time and parks in the first spot available. If none of the k spaces before its preferred parking spot are available, then the car continues east past its preferred spot and parks in the first available spot. If under the parking preference α all cars can pMainRecursionark using this new parking rule, then we say that α is a k-Naples parking function of length n and we denote this set by $PF_{n,k}$. Then $PF_{n,0} = PF_n$, $PF_{n,1}$ is the set of Naples parking functions, and $PF_{n,k-1} \subseteq PF_{n,k}$ for all $0 \leqslant k < n$.

Our first main result provides a recursive formula for the number of k-Naples parking functions of length n.

Theorem 1.1. *If $k, n \in \mathbb{N}$ with $0 \leqslant k \leqslant n - 1$, then the number of k-Naples parking functions of length $n + 1$ is counted recursively by*

$$|PF_{n+1,k}| = \sum_{i=0}^{n} \binom{n}{i} \min((i+1) + k, n+1)|PF_{i,k}|(n-i+1)^{n-i-1}.$$

Given a recurrence, there are well-established ways in which one can develop closed formulas. However, these techniques cannot be applied to the recursive formula in Theorem 1.1 since simplifying the recursion by removing factors yields recurrences that enumerate combinatorial objects for which there are no known formulas.

For example, if we simplify the recursion to $a_{n+1} = \sum_{i=0}^{n} \binom{n}{i} a_i$ with seed values $a_0 = a_1 = 1$, it yields the Bell numbers[2], for which there is no known closed formula.

---

[2]OEIS A000110.

If we incorporate the factor $(n - i + 1)^{n-i-1}$ to the simplified recurrence, then $a_{n+1} = \sum_{i=0}^{n} \binom{n}{i}(n - i + 1)^{n-i-1}a_i$ counts the number of forests of trees on n labeled nodes[3], for which there is also no known closed formula. Lastly, incorporating the term $\min((i+1) + k, n+1)$ into the previous recurrence yields the recursion presented in Theorem 1.1.

In light of the fact that these subsets of the set $PF_{n,k}$ do not have closed formulas for their size, we focus our study on the growth of $|PF_{n,k}^*| := |PF_{n,k} \setminus PF_{n,k-1}|$ as we fix n and increase k from 1 to n, and where $PF_{n,0}^* = \varnothing$. Experimental evidence suggests that $|PF_{n,k}^*|$ is largest when $k = 1$, which corresponds to the number of parking preferences gained by changing the parking rule defining classical parking functions to that defining Naples parking functions. For $n = 25, 50, 75, 100$ and $0 \leqslant k \leqslant n$, we plot the size of $PF_{n,k}^*$ in Figure 2.
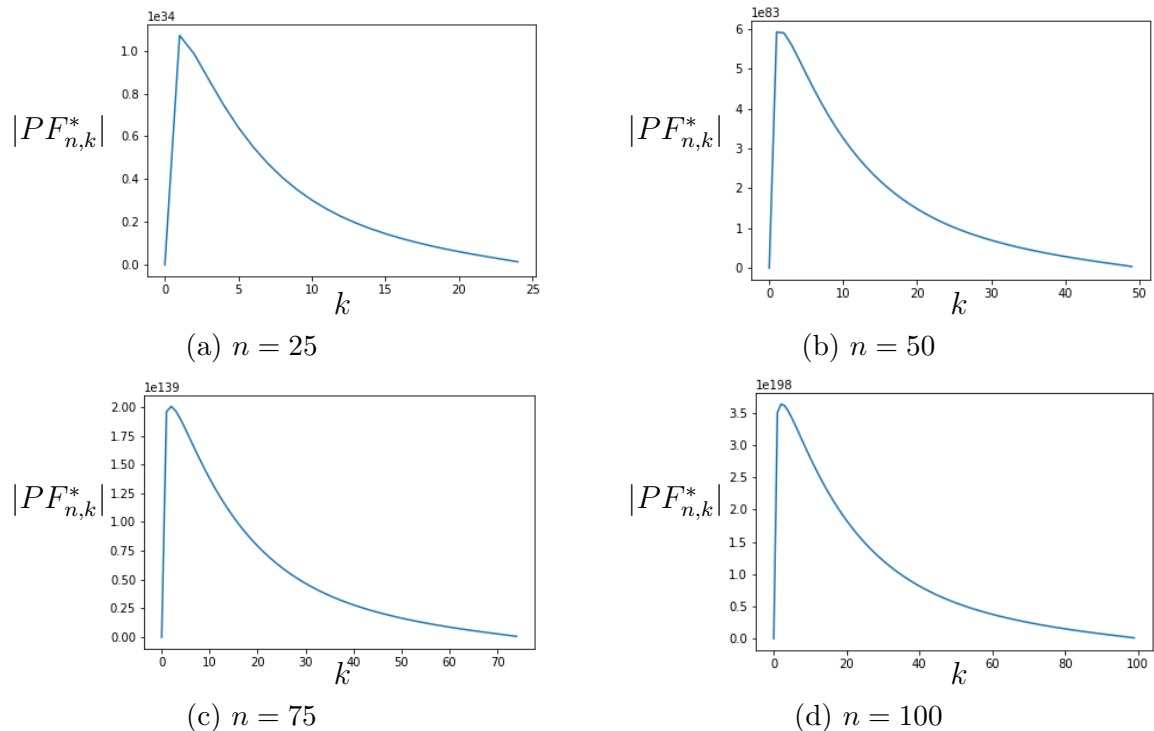

(a) $n = 25$

(b) $n = 50$

(c) $n = 75$

(d) $n = 100$

Figure 2: Plots for $|PF_{n,k}^*|$ for varying values of n and with $1 \leqslant k \leqslant n$. The scale of the y-axis is scaled by a factor of 10^{34}, 10^{83}, 10^{139}, and 10^{198}, when $n = 25, 50, 75,$ and 100, respectively.

Given this observation, Naples parking functions are of particular interest. Our next main result gives a necessary and sufficient condition to characterizing Naples parking functions.

Theorem 1.2. *Fix $n \in \mathbb{N}$. Let $\alpha = (a_1, a_2, \ldots, a_n) \in PP_n$, and define $\mathrm{T} : PP_n \to PP_n$ as $\mathrm{T}(\alpha) = (\tau(a_1), \tau(a_2), \ldots, \tau(a_n))$, where $\tau(a_i)$ is defined*

$$\tau(a_i) = \begin{cases} a_i & \text{if } i = 1, \text{ or if } a_i = 1, \text{ or if } a_i \neq 1 \text{ and } a_i \neq \tau(a_j) \text{ for all } 1 \leqslant j < i \leqslant n \\ a_i - 1 & \text{if } a_i \neq 1 \text{ and } a_i = \tau(a_j) \text{ for some } 1 \leqslant j < i \leqslant n. \end{cases}$$

Then α is a Naples parking function if and only if $\mathrm{T}(\alpha)$ is a parking function.

---

[3]OEIS A001858.

It is known that every rearrangement of the entries of a parking function is also a parking function. However, this is not true for k-Naples parking functions that are not parking functions. Therefore, we study decreasing k-Naples parking functions of length n, those whose entries are in weakly-decreasing order, and give a bijection from this set to a set of decreasing lattice paths of length $2n$, which we call k-lattice paths. These lattice paths are a particular family of signature Dyck paths and we enumerate certain families of them. We note that signature Dyck paths were defined by Cellabos and González D'León, but in general there are no known closed formulas enumerating these combinatorial objects [3].

Theorem 1.3. *If $n, k \in \mathbb{N}$ with $1 \leqslant k \leqslant n$, then the set of decreasing k-Naples parking functions of length n and the set of k-lattice paths of length $2n$ are in bijection.*

This paper is organized as follows. Section 2 gives a precise definition of the k-Naples parking functions, some illustrative examples, and some preliminary results. In Section 3, we prove Theorem 1.1, thereby providing a formula for computing the number of k-Naples parking functions for any length n. Then, in Section 4 and 5, we prove Theorem 1.2 and 1.3, respectively. For the interested reader, we scatter open problems throughout.

2 Background and preliminaries

Given an integer $1 \leqslant k \leqslant n-1$, we consider a new parking rule for the parking preference $\alpha = (a_1, a_2, \ldots, a_n)$. Car c_i drives to its preferred parking spot a_i, and if the spot is occupied, then car c_i first checks back one spot at a time to see if any of the parking spots in the set $A_{i,k} := \{a_i - 1, a_i - 2, \ldots, a_i - k\} \cap [n]$ are available. Note the intersection is present as cars cannot look back past the first parking spot. If any of the spots in $A_{i,k}$ are empty, then c_i parks in the available spot $a_j \in A_{i,k}$ which is closest to its preferred parking spot a_i. If all of the parking spots in the set $A_{i,k}$ are occupied, then c_i proceeds east until it reaches the first unoccupied parking spot after a_i. If under this new parking rule the parking preference α allows all cars to park, then we call α a k-Naples parking function of length n. We denote the set of all k-Naples parking functions of length n by $PF_{n,k}$. We illustrate these definitions below.

Example 2.1. Consider the parking preference $(1, 3, 3, 2)$. Notice that this parking preference is both a parking function and a Naples parking function. However, the order in which the cars park varies, depending on if we are using the classical parking rule or the Naples parking rule. According to the classical parking rule, we have that c_1 parks in the first space, c_2 in the third space, then c_3, finding the third space occupied, continues east and parks in the fourth space, and c_4 parks in its preferred second space. This is illustrated in Figure 3a. In contrast, according to the Naples parking function rule, we have that c_1 parks in the first space, c_2 in the third space, then c_3, finding the third space occupied, looks back a space and parks in the unoccupied second space. Finally, c_4 finds the second space occupied and continues east until it parks in the unoccupied fourth space. This is illustrated in Figure 3b.

We observe that for any parking preference of length n there is a maximum of $n-1$ steps backward that a car can take from its preferred parking space. Moreover, if each

$$\frac{c_1}{1} \quad \frac{c_4}{2} \quad \frac{c_2}{3} \quad \frac{c_3}{4} \qquad\qquad \frac{c_1}{1} \quad \frac{c_3}{2} \quad \frac{c_2}{3} \quad \frac{c_4}{4}$$

$$\text{(a)} \qquad\qquad\qquad\qquad \text{(b)}$$

Figure 3: Illustration of order in which cars with preference $(1,3,3,2)$ park under the classical parking rule (left) and under the Naples parking rule (right).

car can take up to $n-1$ steps backwards then each car is able to check each of the n spaces and all the cars park. Namely,

$$|PF_{n,k}| = |PP_n|, \text{ whenever } k \geqslant n-1. \tag{1}$$

In Table 1, we provide the cardinalities[4] of the sets $PF_{n,k}$ for varying $k \leqslant n$.

n	$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
1	**1**							
2	3	**4**						
3	16	24	**27**					
4	125	203	240	**256**				
5	1,296	2,225	2,731	3,000	**3,125**			
6	16,807	30,067	38,034	42,689	45,360	**46,656**		
7	262,144	484,071	627,405	717,051	773,081	806,736	**823,543**	
8	4,782,969	9,057,316	11,976,466	13,902,752	15,170,350	16,000,823	16,515,072	**16,777,216**

Table 1: The cardinality of $PF_{n,k}$. Numbers in **bold** are n^n, which count the cardinality of $PF_{n,k}$ for $k \geqslant n-1$. The first column, where $k=0$ is $|PF_n| = (n+1)^{n-1}$.

From the sequences in Table 1, the On-line Encyclopedia of Integer Sequences (OEIS) only catalogs the sequences $(PF_{n,0})_{n \in \mathbb{N}}$ and $(PF_{n,n-1})_{n \in \mathbb{N}}$, which are the number of parking functions and the number of parking preferences, respectively. Thus, it appears that many of the sequences associated with k-Naples parking functions have not been studied. However, notice that the difference of the diagonal and subdiagonal in the table arising from the computation of $|PF_{n,n-1}| - |PF_{n,n-2}|$ yields the sequence $1, 3, 16, 125, \ldots$, which is precisely the number of parking functions. In fact there is a bijection between $PF^*_{n,n-1} = PF_{n,k} \setminus PF_{n,k-1}$ and PF_{n-1}, which we discuss in Theorem 2.3. As a consequence of this result, we establish a closed formula for the number of $(n-2)$-Naples parking functions of length n, as presented in Corollary 2.4.

First, in order to formally identify the bijection between $PF^*_{n,n-1}$ and PF_{n-1}, we begin with the following observation about the set $PF^*_{n,n-1}$ of parking preferences that are not k-Naples parking functions for $k < n-1$.

Lemma 2.2. *If $\beta = (b_1, b_2, \ldots, b_n) \in PF^*_{n,n-1}$, then $b_n = n$.*

Proof. By way of contradiction, assume $\beta = (b_1, b_2, \ldots, b_n) \in PF^*_{n,n-1}$ and $b_n \neq n$. All of the cars can park because β is a $(n-1)$-Naples parking function. This implies that when you get to the last car, c_n, only one spot is open. By assumption, c_n's preference satisfies

---

[4]Sequences in Table 1 were computed using https://github.com/andresramos5/Naples-Parking-Function.git.

$1 \leqslant b_n \leqslant n-1$. If c_n arrives to its preferred space and finds it occupied, it first checks backwards. The maximum number of steps back that c_n can take is $(n-1)-1 = n-2 < n-1$. If c_n takes $n-2$ steps back it has checked all the spaces behind its preferred space. Therefore, if the remaining empty space is behind c_n's preferred space then c_n finds it and parks there. If not, c_n can move forward and check all the remaining spaces to find the empty one. Thus, c_n can park with only $n-2$ steps back and $\beta \in PF_{n,n-2}$. This contradicts our assumption that $\beta \in PF_{n,n-1}^* = PF_{n,n-1} \setminus PF_{n,n-2}$. Thus, $b_n = n$. $\qquad \square$

Lemma 2.2 aids in establishing the following result.

Theorem 2.3. *Let* $\alpha = (a_1, a_2, \ldots, a_{n-1}) \in PF_{n-1}$, *and define* $\Psi : PF_{n-1} \to PF_{n,n-1}^*$ *by*

$$\Psi(\alpha) = (\psi(a_1), \ldots, \psi(a_{n-1}), n),$$

where $\psi(a_i) = n+1-a_i$. *Then* Ψ *is a bijection between* PF_{n-1} *and* $PF_{n,n-1}^*$.

Proof. Since $\alpha = (a_1, a_2, \ldots, a_{n-1})$ is a parking function of length $n-1$, we have that $a_i \in [n-1]$ for all i so that $\psi(a_i) \in [n]$. Thus, $\Psi(\alpha) \in PF_{n,n-1} = PP_n$. To verify that $\Psi(\alpha) \notin PF_{n,n-2}$, consider the setup outlined below.

Denote the $n-1$ cars with parking preferences given by α as $c_1, \ldots, c_{n-1}$, and denote the n cars with parking preferences given by $\Psi(\alpha)$ as $d_1, \ldots, d_n$ in order to distinguish between the two. Now, consider Ψ in the following way. For parking function $\alpha \in PF_{n-1}$ arrange for the car c_i to park on a one-way street labeled 1 to n from east to west, where they start driving from the eastern-most space labeled 1 to their desired space a_i and then proceed west if their desired space is occupied. See the red labeling of spaces in Figure 4. Thus, c_i has parking preference $\psi(a_i) = n+1-a_i$ on a lot labeled 1 to n from west to east. See the black labeling of spaces in Figure 4.

$$\overline{\quad 1 \quad} \qquad \overline{\quad 2 \quad} \qquad \overline{\quad 3 \quad} \qquad \cdots \qquad \overline{\quad n-2 \quad} \quad \overline{\quad n-1 \quad} \quad \overline{\quad n \quad}$$
$$\textcolor{red}{n-1} \qquad \textcolor{red}{n-2} \qquad\qquad\qquad \textcolor{red}{3} \qquad\quad \textcolor{red}{2} \qquad\quad \textcolor{red}{1}$$
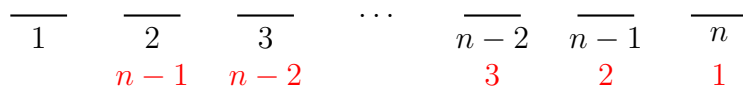
Figure 4: Labeling the parking spaces in two distinct ways.

Since $\alpha \in PF_{n-1}$, the $n-1$ cars park in the red labeled spaces 1 to $n-1$, moving from east to west, or in the black labeled spaces 2 to n, moving west to east. Moreover, car d_i with parking preference $\psi(a_i) = n+1-a_i$ parks in precisely the same spot as c_i, whenever $1 \leqslant i \leqslant n-1$, since d_i proceeds to black space $n+1-a_i$, which is just red space a_i, and then proceeds west to the first available spot if it is unoccupied. Since $\alpha \in PF_{n-1}$, for any $1 \leqslant i \leqslant n-1$ we know that both c_i and d_i never need to check a space further west than the black spot at position 2, i.e. the red spot at position $n-1$, so each car d_i checks at most $n-2$ spaces behind its preferred spot. Thus, cars $d_1, \ldots, d_{n-1}$ park in (black) spaces $2, \ldots, n$. The car d_n must have preference n by Lemma 2.2, implying that the last preference of $\Psi(\alpha)$ is always n and $\Psi(\alpha) \notin PF_{n,n-2}$.

Next, observe that ψ is an involution since

$$(\psi \circ \psi)(a_i) = n+1-(n+1-a_i) = a_i.$$

Thus, Ψ is invertible, which implies it is a bijection. $\qquad \square$

Now we provide a closed formula for the number of $(n-2)$-Naples parking functions of length n.

Corollary 2.4. *If $n \geqslant 2$, then $|PF_{n,n-2}| = n^n - n^{n-2}$.*

Proof. By Theorem 2.3, the set of $(n-1)$-Naples parking functions that are not $(n-2)$-Naples has cardinality $n^{n-2} = ((n-1)+1)^{(n-1)-1} = |PF_{n-1}|$. Moreover, since $PF_{n,n-2}$ and $PF_{n,n-1}^*$ are complimentary in $PF_{n,n-1}$, we have that

$$|PF_{n,n-2}| + |PF_{n,n-1}^*| = |PF_{n,n-1}| = n^n.$$

Therefore, $|PF_{n,n-2}| = n^n - n^{n-2}$ as desired. $\qquad\square$

Having found closed formulas for $|PF_{n,n-1}|$ and $|PF_{n,n-2}|$, in the next section we present a recursive formula to count the number of k-Naples parking functions for all $1 \leqslant k \leqslant n-3$.

3 Counting Naples Parking Functions Recursively

In this section, we begin by introducing a recursive formula for the number of parking functions, first appearing in the work of Konheim and Weiss [5, Equation (2.4), Lemma 1]. For ease of reference, we provide an independent proof of this result and then generalize this recursion so that it counts k-Naples parking functions.

Theorem 3.1. *The number of parking functions of size $n+1$ is recursively counted by the formula*

$$|PF_{n+1}| = \sum_{i=0}^{n} \binom{n}{i}(i+1)^i(n-i+1)^{n-i-1}.$$

Proof. We proceed by counting the number of parking functions of length $n+1$ given that car $n+1$ can park in the spot $i+1$ for $i = 0, 1, 2, \ldots, n$. Let $S \subseteq \{c_1, c_2, \ldots, c_n\}$ consist of the cars parked to the left of the $i+1$ parking space, while the cars that park to the right of the $i+1$ spot consist of the complement of S. Observe that there are $\binom{n}{i}$ ways to select the subset S. The number of ways of assigning parking preferences to the cars in S so that they park before spot $i+1$ is precisely $|PF_i|$. Now, we count the number of ways of assigning parking preferences to the $n-i$ cars found to the right of spot $i+1$ so that they park in the parking spots $i+2$ to $n+1$. This is given by $|PF_{(n+1)-(i+1)}| = |PF_{n-i}|$ since the cars do not park in any of the first $i+1$ spots. Finally, there are $i+1$ possible parking preferences that allow c_{n+1} to park in spot $i+1$. Thus, the number of parking functions of length $n+1$ where car c_{n+1} parks in spot $i+1$ is given by $\binom{n}{i}|PF_i||PF_{n-i}|(i+1)$. Accounting for all possible values of i yields

$$|PF_{n+1}| = \sum_{i=0}^{n} \binom{n}{i}|PF_i||PF_{n-i}|(i+1) = \sum_{i=0}^{n} \binom{n}{i}(i+1)^i(n-i+1)^{n-i-1},$$

as desired. $\qquad\square$

Observe that in order to generalize the recursive formula in Theorem 3.1 to count k-Naples parking functions, we need to modify it by taking into account the new rule that allows cars to look back up to k spots in search for an available one. In this case, if we want car c_{n+1} to park in spot $i+1$, then we must only count the number of parking preferences that allow $n-i$ cars to park in parking spots $i+2$ to $n+1$ without backing up to park in spot $i+1$. Equivalently, we consider introducing an empty parking spot, numbered 0, to the left of 1 and counting the number of k-Naples that would leave that spot open. We refer to this subset of k-Naples parking functions as contained parking functions.

Definition 3.2. The set of **contained parking functions** $B_{n,k}$ is the set of all k-Naples parking functions of length n such that if cars $c_1, \ldots, c_{i-1}$ have already filled spaces $1, \ldots, a_i$, then there is no car c_i with a parking preference $1 \leqslant a_i \leqslant k$.

We call this set the contained parking functions because if you were to introduce more available spots to the ends of the parking lot (before the first spot and/or after the nth spot), the n cars only park in spots $1, \ldots, n$, assuming their parking preferences were between $1, \ldots, n$.

Example 3.3. We let the parking lot be represented by a number line of integers and consider the 2-Naples parking function $\alpha = (4, 4, 2, 3)$, whose cars park as depicted in Figure 5.
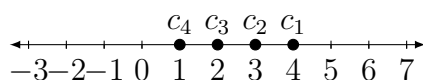

Figure 5: Parking position of cars with parking preference $\alpha = (4, 4, 2, 3)$.

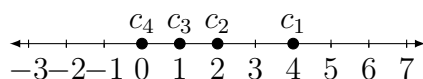If $\beta = (4, 2, 2, 2)$, then the cars park as illustrated in Figure 6.

Figure 6: Parking position of cars with parking preference $\alpha = (4, 2, 2, 2)$

Hence $\beta = (4, 2, 2, 2) \notin B_{4,2}$, because c_4 was able to look back past spot 1 and park in spot 0, leaving spot 3 empty. Thus, the cars' final parking positions are not contained in spots 1 through 4. However, $\beta \in PF_{4,2}$, because under normal conditions it would not check any spot west of 1 and car c_4 would park in spot 3.

With these definitions in hand, we now determine the number of contained parking functions. Our proof adapts Pollak's technique to establish that $|PF_n| = (n+1)^{n-1}$ [4].

Lemma 3.4. *If $n \in N$ and $k \in \{0, 1, \ldots, n\}$, then $|B_{n,k}| = (n+1)^{n-1}$.*

Proof. Consider $\beta \in PP_n$. Each car can check up to k spaces behind their preferred parking spot if it is occupied and only proceeds forward if all the spots they are allowed to check behind them are occupied. Let us arrange these parking spaces clockwise on a circle instead of on a line and introduce a space 0 between 1 and n. Now, if a car's

preferred parking space is occupied, it checks up to k spaces counterclockwise from its preferred parking space and proceeds clockwise if those spots are also occupied. Based on this set up, any parking preference of length n allows all cars to park and leaves one space unoccupied. Observe that the parking preference is an element of $B_{n,k}$ if and only if the cars park in a way that leaves spot 0 unoccupied.

To count the number of ways of assigning n cars parking preferences on the circle, first count the number of ways to assign $n+1$ preferences to n cars, which is $(n+1)^n$. For each parking preference, $\beta = (b_1, \ldots, b_n)$ exactly one "clockwise rotation" of the wheel by an integer j, i.e. the parking preference $(b_1 + j, \ldots, b_n + j) \pmod{(n+1)}$, leaves the spot $n + 1$ unoccupied. Thus, there are $\frac{(n+1)^n}{n+1} = (n+1)^{n-1}$ elements in $B_{n,k}$. $\qquad\square$

Note that Lemma 3.4 implies that the sets $B_{n,k}$ and PF_n are equinumerous. For clarity's sake, it is important to note that the argument used in the proof of Theorem 3.4 cannot be used to count k-Naples parking functions, because in addition to the contained k-Naples parking functions counted in this argument, there are parking functions with k steps back that occupy spot 0 on the circle. For example, $(1, 1, 1) \in PF_{3,1}$, but since cars first check spots behind their preferred parking spot, space 0 on the circle is occupied by the second car. Therefore, there are k-Naples parking functions that are not counted using this argument. Moreover, for small values of n we found that not only are the sets $B_{n,k}$ and PF_n equinumerous, but they also share specific characteristics. To describe these characteristics, we consider $\alpha = (a_1, a_2, \ldots, a_n) \in PP_n$, and for all $1 \leqslant i \leqslant n - 1$, we say

- i is an ascent if $a_i < a_{i+1}$,

- i is a descent if $a_i > a_{i+1}$, and

- i is a tie if $a_i = a_{i+1}$.

Experimentally, the number of ascents, descents and ties in the set $B_{n,k}$ are the same as the number of ascents, descents and ties of PF_n, respectively. The enumeration of descents and ties of parking functions was studied in [7]. These observations lead us naturally to the following open problem.

Problem A. Find a bijection between $B_{n,k}$ and PF_n that preserves the number of ascents, descents and ties.

We now use the set of contained parking functions to give a recursive formula for the number of k-Naples parking functions of length n.

Theorem 1.1. *If $k, n \in \mathbb{N}$ with $0 \leqslant k \leqslant n - 1$, then the number of k-Naples parking functions of length $n + 1$ is counted recursively by*

$$|PF_{n+1,k}| = \sum_{i=0}^{n} \binom{n}{i} \min((i+1) + k, n+1)|PF_{i,k}|(n - i + 1)^{n-i-1}.$$

Proof. As in Theorem 3.1, we now construct a recursion that counts the number of ways that $n + 1$ cars can park given that car c_{n+1} parks in the spot $i + 1$ for $0 \leqslant i \leqslant n$. Let $S \subseteq \{c_1, c_2, \ldots, c_n\}$ consist of the cars parked to the left of the $i + 1$ parking space, while

the cars that park to the right of the $i + 1$ spot consist of the complement of S. There are $\binom{n}{i}$ ways to choose the subset S. Given S, the number of ways of assigning parking preferences to the cars in S which allow them to park in the first i spaces is the number of k-Naples parking functions of length i, $|PF_{i,k}|$. Recall that spot $i + 1$ must remain empty so that car c_{n+1} can park there. Since cars can check up to k spots behind their preferred parking spot, we must be careful to only count the parking preferences for cars in $\{c_1, \ldots, c_n\} \setminus S$ which ensure that they do not park in spot $i+1$. Fortunately, the set of parking preferences we just described is exactly $B_{n-i,k}$, and by Lemma 3.4, we know that $|B_{n-i,k}| = (n-i+1)^{n-i-1}$. Lastly, we count how many possible parking preferences allow car c_{n+1} to park in spot $i+1$. Since car c_{n+1} can check up to k spots behind its preferred spot, a_{n+1}, car c_{n+1} parks in spot $i+1$ only if $1 \leqslant a_{n+1} \leqslant i+1+k$. Also $a_{n+1} \leqslant n+1$, as there are only $n+1$ parking spots. Thus, the number of ways of assigning a parking preference to c_{n+1} so that it parks in spot $i+1$ is $\min((i+1)+k, n+1)$. The result follows from accounting for all possible values of i, which yields

$$|PF_{n+1,k}| = \sum_{i=0}^{n} \binom{n}{i} \min((i+1)+k, n+1)|PF_{i,k}|(n-i+1)^{n-i-1}. \qquad \square$$

Evaluating the equation at $k = 0$ recovers the recurrence from Theorem 3.1.

In this section, we obtained a closed formula for the number of k-Naples parking functions of length n only in the special cases where $k = n - 1$ or $n - 2$ and provided a recursive formula for all other values of $0 \leqslant k \leqslant n - 3$. It remains an open problem to determine closed formulas $|PF_{n,k}|$, but as we discussed in the introduction, such a formula is beyond the scope of our current study. However, we note that

$$|PF_{n,k}| = |PF_n| + |PF_{n,k} \setminus PF_n| = (n+1)^{n-1} + X,$$

and, by Lemma 3.4, we know that $|PF_n| = |B_{n,k}|$. Therefore, we can write $|PF_{n,k}| = |B_{n,k}| + |B_{n,k}^c|$ where $B_{n,k}^c$ is the complement of $B_{n,k}$ in PP_n. Thus, $|PF_{n,k} \setminus PF_n| = |B_{n,k}^c|$. Thus, finding a closed formula for $|B_{n,k}^c|$ is just as difficult as finding a closed formula for $|PF_{n,k}|$. This motivates our next open problem.

Problem B. Find a closed formula to count the number of elements in $B_{n,k}^c$.

4 Characterization of Naples Parking Functions

In this section, we specialize the parameter $k = 1$ and focus our study on the set $PF_{n,1}$, i.e. the set of Naples parking functions of length n. The question of interest is: *Given a parking preference, how can we determine if it is a Naples parking function?* To determine whether a parking preference is a Naples parking function, we define a function which reduces the problem to checking if the image of a Naples parking function is a parking function.

Definition 4.1. Fix $n \in \mathbb{N}$ and let $\alpha = (a_1, a_2, \ldots, a_n) \in PP_n$. We define $\mathrm{T} : PP_n \to PP_n$ as $\mathrm{T}(\alpha) = (\tau(a_1), \tau(a_2), \ldots, \tau(a_n))$, where

$$\tau(a_i) = \begin{cases} a_i & \text{if } i = 1, \text{ or if } a_i = 1, \text{ or if } a_i \neq 1 \text{ and } a_i \neq \tau(a_j) \text{ for all } 1 \leqslant j < i \leqslant n \\ a_i - 1 & \text{if } a_i \neq 1 \text{ and } a_i = \tau(a_j) \text{ for some } 1 \leqslant j < i \leqslant n. \end{cases}$$

We illustrate Definition 4.1 below.

Example 4.2. Let $\alpha = (2, 4, 4, 1) \in PF_{4,1}$. Note $\tau(a_1) = a_1 = 4$, as $i = 1$. Notice that $a_2 = 4 \neq \tau(a_1)$, so $\tau(a_2) = a_2 = 4$. Since $a_3 = 4 \neq 1$ and $\tau(a_2) = a_3$ $\tau(a_3) = a_3 - 1 = 3$. Lastly, $a_4 = 1$, so $\tau(a_4) = 1$. This establishes that $T(\alpha) = (2, 4, 3, 1)$. Note that $T(\alpha) \in PF_4$.

We are now ready to prove Theorem 1.2, which we restate below for ease of reference.

Theorem 1.2. *Fix $n \in \mathbb{N}$. Let $\alpha = (a_1, a_2, \ldots, a_n) \in PP_n$, and define $\mathrm{T} : PP_n \to PP_n$ as $\mathrm{T}(\alpha) = (\tau(a_1), \tau(a_2), \ldots, \tau(a_n))$, where $\tau(a_i)$ is defined*

$$\tau(a_i) = \begin{cases} a_i & \text{if } i = 1, \text{ or if } a_i = 1, \text{ or if } a_i \neq 1 \text{ and } a_i \neq \tau(a_j) \text{ for all } 1 \leqslant j < i \leqslant n \\ a_i - 1 & \text{if } a_i \neq 1 \text{ and } a_i = \tau(a_j) \text{ for some } 1 \leqslant j < i \leqslant n. \end{cases}$$

Then α is a Naples parking function if and only if $\mathrm{T}(\alpha)$ is a parking function.

Proof. We first show that if $\alpha \in PF_{n,1}$, then $\mathrm{T}(\alpha) \in PF_n$. Suppose $\alpha = (a_1, a_2, \ldots, a_n) \in PF_{n,1}$ and $\mathrm{T}(\alpha) = (b_1, b_2, \ldots, b_n)$. By Definition 4.1, we know that for each $1 \leqslant i \leqslant n$, $b_i = a_i$ or $b_i = a_i - 1$. In particular, if some car c_i has preferred spot a_i and that spot is taken by some car c_j, with $1 \leqslant j < i$, then $b_i = a_i - 1$. Otherwise, we have $b_i = a_i$. Since c_i can park using the Naples parking rule, then there exists a spot q with $a_i - 1 \leqslant b_i \leqslant q \leqslant n$ that is unoccupied. In other words, there must be an empty spot somewhere between spots $a_i - 1$ and n in order for c_i to park. Because $a_i - 1 \leqslant b_i$, the new preference b_i ensures that the car finds an empty spot to park in, which is either at position b_i or somewhere ahead of it. Thus, c_i is able to park using the original parking rule. Since i is arbitrary, each car c_i with preference b_i can park for $1 \leqslant i \leqslant n$ using the original parking rule.

To show that $\mathrm{T}(\alpha) \in PF_n$ implies $\alpha \in PF_{n,1}$ we prove the contrapositive. That is, if $\alpha \notin PF_{n,1}$ then $\mathrm{T}(\alpha) \notin PF_n$. Let $\mathrm{T}(\alpha) = (b_1, b_2, \ldots, b_n)$, where $\alpha = (a_1, a_2, \ldots, a_n) \notin PF_{n,1}$. As above, $b_i = a_i$ or $b_i = a_i - 1$. Since $\alpha \notin PF_{n,1}$ there exists a car c_i that cannot park using the Naples parking rule. That means that there does not exists an available spot q satisfying $a_i - 1 \leqslant b_i \leqslant q \leqslant n$. Moreover, since none of these spots are available for parking using the Naples parking rule, they are also not available when parking using the original parking rule. Thus, $\mathrm{T}(\alpha) \notin PF_n$. $\qquad\square$

With the complete characterization of Naples parking functions complete, we now study their connection to Dyck paths.

5 Connections to Decreasing Lattice Paths

In this section, we introduce k-Lattice paths, a generalization of Dyck paths, and give a bijection between these objects and decreasing k-Naples parking functions. This result exploits the classical result which gives a correspondence between Dyck paths and decreasing[5] parking functions. We end the section by connecting our main result, Theorem

---

[5]The original proof considers increasing parking functions, but the bijection holds under a slight change of indices for the decreasing parking functions.

1.3, to signature Catalan objects, as presented in the work of Cellabos and González D'León [3].

In what follows, we consider decreasing rearrangements of k-Naples parking functions, as increasing rearrangements of k-Naples parking functions are not necessarily k-Naples. For example, $(4, 1, 4, 3), (4, 4, 3, 1) \in PF_{4,1}$, but $(1, 3, 4, 4) \notin PF_{4,1}$. Therefore, it is more natural to consider Dyck paths drawn from $(0, n)$ to $(n, 0)$ using east and south steps. We present our formal definition below.

Definition 5.1. For a given $n \in \mathbb{N}$, a **Dyck path** of length $2n$ is a lattice path from $(0, n)$ to $(n, 0)$ consisting of n steps by $(1, 0)$ east and n steps by $(0, -1)$ south such that the path never goes above the line $y = n - x$. For any south step, the number of south steps proceeding it is larger than the number of east steps preceding it. We denote the set of all Dyck paths of length $2n$ as LP_n.

We now describe the bijection between decreasing parking functions and Dyck paths. Recall that a decreasing parking function is one whose entries are written in weakly-decreasing order. Specifically, if $\alpha = (a_1, \ldots, a_n) \in PF_n$ is a decreasing parking function, then the corresponding lattice path has an east step $(i - 1, a_i - 1)$ to $(i, a_i - 1)$ at height $a_i - 1$ for each $1 \leqslant i \leqslant n$, and south steps connecting these east steps so that the result is a connected path from $(0, n)$ to $(n, 0)$. The fact that α is a decreasing parking function implies that $a_i \leqslant n - i + 1$, hence the corresponding lattice path does not cross the line $y = n - x$.

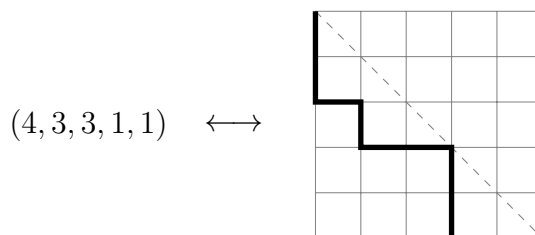$$(4, 3, 3, 1, 1) \quad \longleftrightarrow$$

Figure 7: Dyck path corresponding to $\alpha = (4, 3, 3, 1, 1)$.

Example 5.2. Let $\alpha = (4, 3, 3, 1, 1)$ and note that its associated Dyck path has one east step at height 3, two east steps at height 2, and two east steps at height 0. Figure 7 illustrates the corresponding Dyck path for α.

We now consider a generalization of Dyck paths, which we call k-lattice paths.

Definition 5.3. If $n, k \in \mathbb{N}$ with $0 \leqslant k \leqslant n - 1$, then a **$k$-lattice path** of length $2n$ is a lattice path from $(0, n)$ to $(n, 0)$ consisting of n steps east by $(1, 0)$ and n steps south by $(0, -1)$ such that the path does not cross the line $y = n - x + k$. We denote the set of all k-lattice paths of length $2n$ as $LP_{n,k}$.

Notice that, $LP_{n,0} = LP_n$, which is the set of Dyck paths of length $2n$. Thus, k-lattice paths are a generalization of Dyck paths. Next, we present our main result, which establishes a bijection between decreasing k-Naples parking functions and k-lattice paths. Since it is well-known that there is a bijection between LP_n and decreasing parking functions of length n, in what follows, we only consider the case where $1 \leqslant k \leqslant n - 1$.

Theorem 1.3. *If $n, k \in \mathbb{N}$ with $1 \leqslant k \leqslant n$, then the set of decreasing k-Naples parking functions of length n and the set of k-lattice paths of length $2n$ are in bijection.*

Proof. To establish this result, it suffices to show that given a decreasing k-Naples parking function we can construct a k-lattice path, and given a k-lattice path there is a corresponding decreasing k-Naples parking function.

Suppose that $\alpha = (a_1, a_2, \ldots, a_n)$ is a decreasing k-Naples parking function. As in the parking function case, from α we construct the k-lattice path of length $2n$ with east steps $(i-1, a_i - 1)$ to $(i, a_i - 1)$, which we denote $LP(\alpha)$.

We need only show that $a_i \leqslant \min(n, n+k+1-i)$ holds for all $1 \leqslant i \leqslant n$, as this implies that $LP(\alpha)$ is a k-lattice path. Suppose that there is some i such that $a_i > \min(n, n+k+1-i)$ to obtain a contradiction. For $i = 1, 2, \ldots, k+1$, note that $\min(n, n+(k+1)-i) = n$, and because $\alpha \in PF_{n,k}$ it cannot be that $a_i > n$.

On the other hand, if $k+1 < i \leqslant n$, then $\min(n, n+k+1-i) = n+k+1-i$, and lets assume that $a_i > n+k+1-i$. Since α is in decreasing order we know that $a_j \geqslant a_i > n+k+1-i$ for all $1 \leqslant j < i$. The most optimal parking preference is $a_j = n+k+2-i$ for all $1 \leqslant j < i$, as this maximizes the number of parking positions cars $c_1, \ldots, c_j$ can occupy. That is, it leaves the most open spots to the right of position $n+k+2-i$. In this case, cars $c_1, c_2, \ldots, c_{k+1}$ park in positions $n+k+2-i, n+k+1-i, \ldots, n+2-i$, respectively. Then, cars $c_{k+2}, c_{k+3}, \ldots, c_{i-1}$ first go to their preferred parking spot, namely $n+k+2-i$, finding it occupied they back up and all of the k prior spots are also full. Thus, these cars go forward and occupy the last $i - k - 2$ spots numbered $n+k+3-i$ to n. Then car c_i, arriving to its preferred position, again $n+k+2-i$, finding it occupied backs up and also finds all k spots behind full. It then moves forward and finds all remaining spots taken. Thus, c_i is unable to park contradicting the assumption that $\alpha \in PF_{n,k}$.

We now go from an arbitrary k-lattice path to a decreasing k-Naples parking function. Given a lattice path $L \in LP_{n,k}$, we know this path starts at $(0, n)$, ends at $(n, 0)$, and stays below the line $y = n - x + k$. Suppose the east steps of L occur from $(i-1, a_i-1)$ to (i, a_i-1), then by definition $a_i \leqslant \min(n, n+k+1-i)$ for all $1 \leqslant i \leqslant n$. Construct the parking preference $\alpha = (a_1, a_2, \ldots, a_n)$. Note that the construction of α guarantees that α is in decreasing order. It remains for us to show that $\alpha \in PF_{n,k}$. That is, we check that for all $i \in [n]$, car c_i can park under the k-Naples parking rules.

First, observe that the first $k+1$ cars can always park, since they can back up to k positions, see Equation (1). Now for $i > k+1$ we split into two cases: $a_i \leqslant k$ and $a_i > k$.

Case 1: Suppose $i > k+1$ and $a_i \leqslant k$. If one of the spots between 1 and k is unoccupied, then c_i parks there. Instead, if all of the parking spots between 1 and k are occupied, this means that of the $i-1$ cars that have parked, k of them have occupied the first k spots, while the remaining $i - (k+1)$ cars occupied some spots numbered between $k+1$ and n. Thus, there are less cars parked to the right of spot k, than there are parking spots between $k+1$ and n. Thus, c_i parks in the leftmost available spot between $k+1$ and n.

Case 2: Suppose $i > k+1$ and $a_i > k$. If spot a_i or a spot up to k steps behind is unoccupied then c_i can park. Otherwise, spots $a_i - k, a_i - k + 1, \ldots, a_i$ are occupied. Now, since $a_i \leqslant n+k+1-i$, we have that there are $n - a_i \geqslant n - (n+k+1-i) = i - (k+1)$ spots to the right of a_i. By assumption, spots $a_i - k$ through a_i are occupied by $k+1$ of the $i - (k+1)$ cars that parked before c_i so that $i - (k+2)$ cars have parked in the at

least $i - (k+2)$ spots after a_i. This leaves a remaining open spot between a_{i+1} and n in which c_i can park. Since i was arbitrary, we have established that all cars c_i can park for all $1 \leqslant i \leqslant n$. Thus, $\alpha \in PF_{n,k}$. $\qquad\square$

Now we provide a connection between k-lattice paths and signature Dyck paths. In a recent paper by Cellabos and González D'León [3], they introduce the concept of signature Dyck paths, defined by a vector $s = (s_1, s_2, \ldots, s_\ell) \in \mathbb{N}^\ell$. The signature s defines a ribbon and an s-Dyck path is a lattice path that lies on or above the ribbon. To describe the ribbon, we construct a grid of dimensions $\ell \times [(\sum_{i=1}^\ell s_i - 1) + 1]$. We number the boxes in this grid from bottom to top, calling each row a level i with $1 \leqslant i \leqslant \ell$, and on each level we number the boxes left to right from 1 to $(\sum_{i=1}^\ell s_i - 1) + 1$. At each level we shade a specific set of boxes. Begin by shading the boxes $1, 2, \ldots, s_1$ at level 1. Then, for $2 \leqslant i \leqslant \ell$, shade the boxes numbered $(\sum_{j=1}^{i-1} s_j - 1) + 1$ to $(\sum_{j=1}^{i} s_j - 1) + 1$ at level i. Figure 8, illustrates the ribbon when $s = (3, 2, 5, 1, 1)$, along with an s-Dyck path in blue, and a lattice path that is not an s-Dyck path in red.

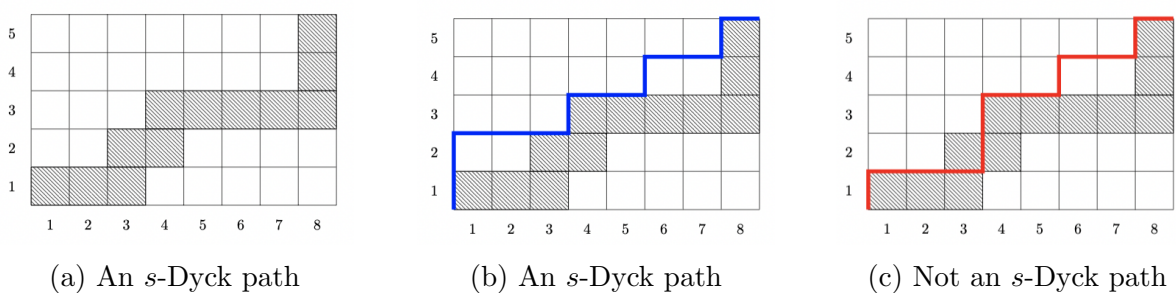(a) An s-Dyck path   (b) An s-Dyck path   (c) Not an s-Dyck path

Figure 8: The ribbon corresponding to the signature $s = (3, 2, 5, 1, 1)$, and two lattice paths: one an s-Dyck path (blue path) and that is not an s-Dyck path (red path).

In our work, we consider a horizontal reflection of s-Dyck paths so that our paths are decreasing, rather than increasing. In this way, k-lattice paths of length $2n$ are s-Dyck paths with signature

$$s = (k+1, \underbrace{2, 2, \ldots, 2}_{n-k}, \underbrace{1, 1, \ldots, 1}_{k}) \tag{2}$$

of length $2(n+1)$. In Figure 9, we illustrate the signature for a 3-lattice path of length 14, and note that any lattice path begins with a south step from $(0, 7)$ to $(0, 6)$ and ends with an east step from $(6, 0)$ to $(7, 0)$.

Theorem 1.3 along with the resulting sequences in Table 2 give formulas for the number of s-Dyck paths with signatures as given in Equation (2), for the special cases $k = 1, 2$, and 3. We note that formulas for other values of k are unknown.

Remark 5.4. By Theorem 1.3 we know that for $k+1 \leqslant n$ the decreasing parking function

$$(\underbrace{n, \ldots, n}_{k+1}, n-1, n-2, \ldots, k+1) \in PF_{n,k}.$$

Moreover, for fixed n and $k < n - 1$ we can use the above decreasing parking function to generate a subset of $PF^*_{n,k+1}$ and obtain a rough lower bound on $|PF^*_{n,k}|$. Specifically,
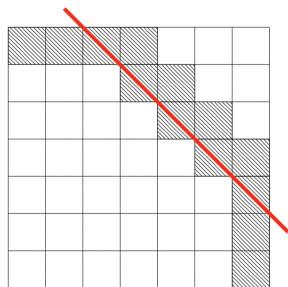
Figure 9: Illustrating the possible locations for 3-lattice paths of length 12, which begin at $(0,6)$, end at $(6,0)$, and must lie below the red line given by $y = 6 - x + 3$. This corresponds to s-Dyck paths of length 14, which begin at $(0,7)$, end at $(7,0)$, and must lie on or below the signature $s = (4, 2, 2, 2, 1, 1, 1)$.

Subset of k-Naples parking functions	OEIS	Sequence	Formula
$|PF_{n,1}^d|$ with $n \geqslant 2$	A000245	$3, 9, 28, 90, 297, \ldots$	$\frac{3(2n)!}{(n+2)!(n-1)!}$
$|PF_{n,2}^d|$ with $n \geqslant 3$	A026016	$10, 34, 117, 407, \ldots$	$\binom{2(n-1)}{n} - \binom{2(n-1)}{(n+3)}$
$|PF_{n,3}^d|$ with $n \geqslant 4$	A026026	$35, 125, 451, 1638, \ldots$	$\binom{2n-1}{n-1} - \binom{2n-1}{n-5}$

Table 2: Known integer sequences related to enumerating decreasing k-Naples parking functions, which we denote as $PF_{n,k}^d$.

given the decreasing parking function $(\underbrace{n, \ldots, n}_{k+1}, n - 1, n - 2, \ldots, k + 1) \in PF_{n,k}$, we may add 1 to any of the $n - k - 1$ entries $n - 1, n - 2, \ldots, k + 1$ and still obtain a valid parking preference for a parking lot with n spaces. There are $2^{n-k-1} - 1$ distinct ways of doing so, which excludes the option of not adding one anywhere. Each of the $2^{n-k-1} - 1$ possibilities corresponds to a parking preference, which is not equal to the original. By Theorem 1.3, each of these new parking preferences are elements of $PF_{n,k+1}^*$. Thus, we have the lower bound

$$2^{n-k-1} - 1 \leqslant |PF_{n,k+1}^*|.$$

This quantity grows very rapidly depending on the size of $n - k$, and we know that we have the strict inequality $|PF_{n,k}| < |PF_{n,k+1}|$, since by assumption $k < n - 1$. It would be interesting to study the asymptotics of $|PF_{n,k}|$ for specific values of k. Hence, we ask:

Problem C. Are there k for which $|PF_{n,k}|$ and $|PF_{n,k+1}|$ are asymptotically equivalent?

Note that for $n \geqslant 2$, Corollary 2.4 states that $|PF_{n,n-2}| = n^n - n^{n-2}$ and we know $|PF_{n,n-1}| = n^n$. Hence, they are asymptotically equivalent to n^n. This motivates the following.

Problem D. Are there $k > 2$ for which $|PF_{n,n-k}|$ and $|PF_{n,n-k+1}|$ are not asymptotically equivalent?

Looking at Table 1, Problem C asks if any of the verticals are asymptotically equivalent, whereas Problem D asks if any of the diagonals are **not** asymptotically equivalent.

5.1 Rearrangements of k-Naples parking functions

We begin by illustrating that not all rearrangements of k-Naples parking functions are k-Naples parking functions.

Example 5.5. Let $\alpha = (7, 7, 7, 7, 5, 2, 2) \in PP_7$. We now verify that α is an element of $PF_{7,3}$. First, c_1 parks in its preferred parking spot 7. Then, c_2 backs up one space and parks in position 6, c_3 backs up two spaces and parks in position 5, c_4 backs up three spaces and parks in position 4, c_5 backs up two spaces and parks in position 3. Next, c_6 parks in its preferred parking space 2, while the last car, c_7, has to back up one space to park in position 1. The filled parking lot based on α is illustrated in Figure 10.

$$\frac{c_7}{1} \quad \frac{c_6}{2} \quad \frac{c_5}{3} \quad \frac{c_4}{4} \quad \frac{c_3}{5} \quad \frac{c_2}{6} \quad \frac{c_1}{7}$$

Figure 10: Illustrating the parking order for the 3-Naples parking function $(7, 7, 7, 7, 5, 2, 2)$.

Now, notice that in the rearrangement $\beta = (5, 7, 7, 7, 7, 2, 2)$ of α, c_1 parks at position 5, c_2 parks at position 7, c_3 backs up one space and parks at position 6, and c_4 backs up three spaces and parks at position 4. Now c_5 finds its preferred space and the three preceding occupied. Additionally, when it checks forward, there are no available spaces and c_5 cannot park. Thus, $\beta \notin PF_{7,3}$.

Characterizing when a rearrangement of a k-Naples parking function is another k-Naples parking function remains an open problem. We state this formally below.

Problem E. Characterize and enumerate which rearrangements of decreasing k-Naples parking functions are also elements of $PF_{n,k}$.

In the case where $k = 1$ we conjecture the following.

Conjecture 5.6. If there is only one corner above the line $y = n - x$ then that parking preference and all of its rearrangements are elements of $PF_{n,1}$.

Another problem of interest would be to consider the case where there are m cars in a parking lot with n spaces, where $m < n$, and cars are allowed to back up k spots. This idea leads naturally to asking about the enumeration of k-Naples (m, n)-parking functions, where the $k = 0$ case is enumerated by $(n+1-m)(n+1)^{m-1}$ [5, Lemma 2]. We end by pointing the interested reader to the work of Carlson, Christensen, Harris, Jones, and Ramos Rodríguez, which provides many new directions for further investigation on generalized parking functions [2].

Acknowledgements

References

[1] Alyson Baumgardner, *The naples parking function*, Honors Contract-Graph Theory, Florida Gulf Coast University, 2019.

[2] Joshua Carlson, Alex Christensen, Pamela E. Harris, Zakiya Jones, and Andres Ramos Rodriguez, *Parking functions: Choose your own adventure*, Preprint, https://arxiv.org/pdf/2001.04817.pdf.

[3] Cesar Ceballos and Rafael S. González D'León, *Signature catalan combinatorics*, Journal of Combinatorics **10** (2019), 725–773.

[4] Dominique Foata and John Riordan, *Mappings of acyclic and parking functions*, Aequationes Math. **10** (1974), 10–22. MR0335294

[5] Alan G. Konheim and Benjamin Weiss, *An occupancy discipline and applications*, Siam Journal on Applied Mathematics - SIAMAM **14** (196611).

[6] Ronald Pyke, *The supremum and infimum of the poisson process*, Ann. Math. Statist. **30** (195906), no. 2, 568–576.

[7] Paul R. F. Schumacher, *Descents in parking functions*, J. Integer Seq. **21** (2018), no. 2, Art. 18.2.3, 8. MR3779772

[8] Catherine H. Yan, *Generalized parking functions, tree inversions, and multicolored graphs*, Adv. in Appl. Math. **27** (2001), no. 2-3, 641–670. Special issue in honor of Dominique Foata's 65th birthday (Philadelphia, PA, 2000). MR1868985

[9] _____, *Parking functions*, Handbook of enumerative combinatorics, 2015, pp. 835–893.

Corrigendum added 2 April 2020.

Corrigendum − Added February 8, 2021

1. In Definition 5.3, we implicitly assumed that the paths we considered always began with a south step, as we specified in Definition 5.1. The new version of Definition 5.3 is now:

   Definition 5.3. If $n, k \in \mathbb{N}$ with $0 \leqslant k \leqslant n - 1$, then a **$k$-lattice path** of length $2n$ is a lattice path from $(0, n)$ to $(n, 0)$ consisting of n steps east by $(1, 0)$ and n steps south by $(0, -1)$ such that the path does not cross the line $y = n - x + k$ and always begins with a south step. We denote the set of all k-lattice paths of length $2n$ as $LP_{n,k}$.

   Note that Theorem 1.3 holds as stated without modifications since we assumed this updated definition from the onset.

2. In Table 2, the last column: The correct formula for the number of k-Naples parking functions in the case $k = 2$ should be given by $\binom{2n-1}{n} - \binom{2n-1}{n+3}$.

3. We thank Prof. Dr. Volker Strehl, who pointed the above two discrepancies and also for sharing the work of Flajolet (Combinatorial aspects of continued fractions. Discrete Math. 32 (1980), no. 2, 125–161) as a good starting point to explore this area further. More importantly, he pointed out that using what is known as "the reflection principle" one is able to provide binomial formulas for all values of k. Namely,

$$|PF_{n,k}^d| = \binom{2n - 1}{n} - \binom{2n - 1}{n + k + 1}.$$