Rearrangements of Naples parking functions

Zakiya Jones, Christo Keller,
Andrés Ramos, Eunice Sukarto

Faculty Mentor: Laura Colmenarejo

2020 Advancing Inquiry/Inclusion in Mathematics Undergraduate Program

Outline

Naples parking functions

Decreasing k-Naples

Rearrangements of k-Naples

Permutations and Parking Functions

Parking functions

- The Naples parking functions were defined in Baumgardner 2019. [Bau19]
- These objects generalize the parking rule by allowing cars to try to park one spot behind if their parking spot is already taken when they check it.
- The k-Naples parking functions from Christensen et al 2020 generalize this further by allowing cars to try to park k spots behind from right to left. [CHJ+20]

Example Naples parking function

Example: $(3,2,2)$ is Naples but not classical.

Parking function (0-Naples):
empty empty $c_1 \to$ empty c_2 $c_1 \to$ empty c_2 c_1 c_3

Naples function (1-Naples):
empty empty $c_1 \to$ empty c_2 c_1 $\to c_3$ c_2 c_1

What we know about $PF_{n,k}$

- Enumeration of k-Naples parking functions

Theorem [CHJ+20]

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

- Connection to Dyck Paths

Theorem [CHJ+20]

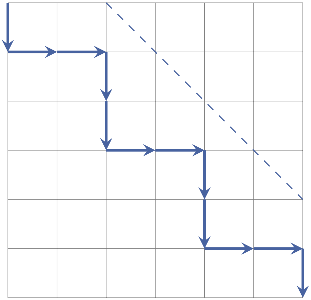
If $n, k \in \mathbb{N}$ with $1 \leq k \leq n$, then the set of decreasing k-Naples parking functions of length n and the set of k-lattice paths of length $2n$ are in bijection.

What we know about $PF_{n,k}$

- Given a tuple, $\alpha = (a_1, a_2, \ldots, a_n)$, draw a path from $(0, n)$ to $(n, 0)$ such that a_i corresponds to the i^{th} east step
- $\alpha \in PF^d_{n,k}$ if and only if its corresponding path does not cross the line $y = n - x + k$
- Note that we denote the set of non-increasing k-Naples parking functions as $PF^d_{n,k}$
- If a parking preference is k-Naples, then so is its decreasing rearrangement.

What we know about $PF_{n,k}$

For example, we can associate the 2-Naples parking function $\alpha = (6, 6, 4, 4, 2, 2)$ to the following 2-lattice path.



A Closed Formula for $|PF^d_{n,k}|$

$|PF^d_{n,k}|$ = number of lattice paths from $(0, n)$ to $(n, 0)$ not going above the line $y = n - x + k$.

Theorem [Bollobás (2006), CHJKRSV (2020)]

$$|PF^d_{n,k}| = \binom{2n-1}{n} - \binom{2n-1}{n+k+1}$$

- Problem 61 in "The Art of Mathematics: Coffee Time in Memphis" (Bollobás)
- *Our proof* uses **induction** on k and **inclusion-exclusion**

$$P_{n,k} = P_{n+1,k-1} - P_{n,k-1} - P_{n,k-2} - P_{n+1,k-2} + P_{n,k-2} + P_{n,k-3} + P_{n,k-2}$$

k-Naples q-analogues

We can take the q-analogue of the number of decreasing k-Naples PFs

$$|\mathsf{PF}^d_{n,k}|_q = \begin{bmatrix} 2n-1 \\ n \end{bmatrix}_q - \begin{bmatrix} 2n-1 \\ n+k+1 \end{bmatrix}_q \in \mathbb{Z}[q]$$

It's an **open problem** to discover which statistic has this distribution over the decreasing k-Naples parking functions, that is for what stat : $\mathsf{PF}^d_{n,k} \to \mathbb{N}$

$$\sum_{\pi \in \mathsf{PF}^d_{n,k}} q^{\mathsf{stat}(\pi)} = |\mathsf{PF}^d_{n,k}|_q$$

Algorithm for Constructing $PF^d_{n,k}$ [CHJKRSV (2020)]

$PF^d_{n,k}$ is the union of the following **two disjoint sets** according to bumps:
1. The first set is constructed recursively by taking the subset of $PF^d_{n-1,k}$ and adding an entry $1, 2, 3, \ldots, n$ in front of each $n-1$-tuple such that the entries remain in decreasing order. **This is precisely the set with no bump in the second entry.**

- For example, the decreasing $PF_{3,1}$ are: $(111), (211), (221), (222), (311), (321), (322), (331)$, and (332).
- By putting $1, 2, 3$, or 4 in front of each of these we get a subset decreasing $PF_{4,1}$

Algorithm for Constructing $PF^d_{n,k}$ [CHJKRSV (2020)]

2. The second set is constructed as follows: Let $a_1 = a_2 = n$, $a_3 = n+k-2, n+k-3, \dots, 1$, and $a_4 = n+k-3, n+k-4, \dots, 1, \dots, a_n = k+1, k, k-1, \dots, 1$. Now, take all possible combinations of $a_3, a_4, \dots, a_n$ such that $a_1 \geq a_2 \geq \cdots \geq a_n$. **This is exactly the set of $PF^d_{n,k}$ with a bump in the second entry.**

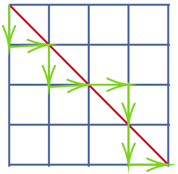
- For example, this subset of $PF^d_{4,1}$ are all possible ways to construct a 4-tuple such that the first two entries are 4, the third entry is $1, 2$, or 3, and the fourth entry is 1 or 2, such that all of the entries are in non-increasing order.
- These are $(4432), (4431), (4422), (4421)$, and (4411)

About reorderings of *k*-Naples

- We want to characterize *k*-Naples Parking Functions.

Example

Consider $\alpha = (4, 3, 3, 1) \in PF_{n,1}$.



Our result

Conjecture [CHJLRRR (2020)]

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

Proposition [CHJKRSV (2020)]

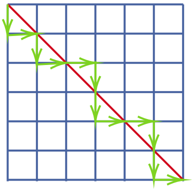
Let $\alpha = (a_1, \ldots, a_n)$ be a decreasing Naples parking function. Then $\pi(\alpha) \in PF_{n,1}$ for all $\pi \in S_n$ if and only if α has no adjacent bumps.

aim UP

Example

Example

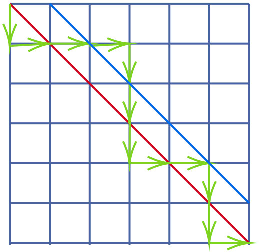
Consider $\alpha = (6,5,5,3,3,1) \in PF_{6,1}$. Then every rearrangements of $\alpha \in PF_{6,1}$.



General rules for rearranging bumps

Can we find rules for rearranging parking functions with more complicated bumps?

Let $\alpha = (6, 6, 6, 3, 3, 1) \in PF_{6,2}$.



Conjectures and Takeaways

Conjecture [CHJKRSV (2020)]

Suppose $\alpha \in PF^d_{n,1}$ with bump multisets $B_{i_1}, \ldots, B_{i_m}$ where each car c_{i_j} is a lucky car. Then $\pi(\alpha) \in PF_{n,1}$ if and only if $\pi(\alpha) \in B_{i_1} \sqcup \cdots \sqcup B_{i_m} \sqcup a_l$ for all a_l not in any bump multiset.

Takeaways

1. Shuffling non-disjoint sets is hard
2. We're just scratching the surface

A Map to... Permutations

Consider $\varphi : PF_n \to S_n$ mapping a parking function to the resulting parking arrangement.

It is mentioned here

https://www.findstat.org/StatisticsDatabase/St000280/

How many parking functions map to $s_1 \ldots s_n \in S_n$?

Proposition [CHJKRSV (2020)]

$$|\varphi^{-1}(s_1 \ldots s_n)| = \prod_{i=1}^{n} \ell_i$$

where ℓ_i is the length of the longest subsequence $s_j \ldots s_i$ such that $s_t \leq s_i$ for all $j \leq t \leq i$.

Example: How many parking functions map to 23514?

$$
\begin{aligned}
s_1 &= 2 \rightsquigarrow 2 && \Rightarrow \ell_1 = 1\\
s_2 &= 3 \rightsquigarrow 23 && \Rightarrow \ell_2 = 2\\
s_3 &= 5 \rightsquigarrow 235 && \Rightarrow \ell_3 = 3\\
s_4 &= 1 \rightsquigarrow 1 && \Rightarrow \ell_4 = 1\\
s_5 &= 4 \rightsquigarrow 14 && \Rightarrow \ell_5 = 2
\end{aligned}
$$

So $1 \times 2 \times 3 \times 1 \times 2 = 12$.

An Expression for $|PF_n|$

Theorem [CHJKRSV (2020)]

$$\sum_{s \in S_n} \left(\prod_{i=1}^{n} \ell_i \right) = |PF_n| = (n+1)^{n-1}$$

Note: ℓ is with respect to each permutation

What about *k*-Naples?

Consider $\varphi_k : PF_{n,k} \to S_n$ mapping a k-Naples parking function to the resulting parking arrangement under the k-Naples rule.

Caution!

$$(4,2,2,1,2) \in PF_5 \text{ vs. } (4,2,2,1,2) =\in PF_{5,2}$$

$$\varphi(4,2,2,1,2) = 42315 \neq \varphi_2(4,2,2,1,2) = 32415$$

How many k-Naples parking functions map to $s_1 \ldots s_n \in S_n$?

Proposition [CHJKRSV (2020)]

$$|\varphi_k^{-1}(s_1 \ldots s_n)| = \prod_{i=1}^{n} \ell_k(i)$$

i_{left} = length of longest subsequence $s_j \ldots s_{i-1}$ such that $s_t \leq s_i$ for all $j \leq t < i$
i_{right} = length of longest subsequence $s_i \ldots s_r$ where $r \leq i + k$, such that $s_t \leq s_i$ for all $i \leq t \leq i + k$.

$$\ell_k(i) = \begin{cases} i_{\text{left}} + i_{\text{right}} & \text{if } i_{\text{left}} = i - 1 \\ \max(i_{\text{left}} - k, 0) + i_{\text{right}} & \text{if } i_{\text{left}} < i - 1 \end{cases}$$

An Expression for $|PF_{n,k}|$

Theorem [CHJKRSV (2020)]

$$|PF_{n,k}| = \sum_{s \in S_n} \left(\prod_{i=1}^{n} \ell_k(i) \right)$$

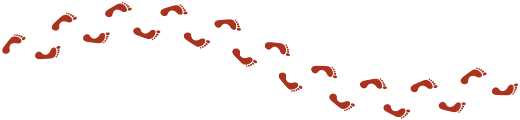
Note: ℓ_k is with respect to each permutation

Compare to Christensen et al. 2020

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


Hard to swallow pills
You will never have a closed formula for the number of k-Naples parking functions.
54
543

Thank you!



References

A. Adeniran, S. Butler, G. Dorpalen-Barry, P. E. Harris, C. Hettle, Q. Liang, J. L. Martin, and H. Nam.
Enumerating parking completions using join and split.
The Electronic Journal of Combinatorics, 27(2):#P2.44, 2020.

A. Adeniran.
Gončarov polynomials, partition lattices and parking sequences.
Doctoral dissertation, Texas A&M University, 2020.
Available electronically from http://hdl.handle.net.

A. Adeniran and C. Yan.
Gončarov polynomials in partition lattices and exponential families.
Advances in Applied Mathematics, 2020.

A. Adeniran and C. Yan.
On increasing and invariant parking sequences.
2020.
https://arxiv.org/pdf/2005.04759.pdf.