Homomesy & Parking Functions

Marvin Gandhi & Cyrus Young

Mentored by Dr. Pamela E. Harris & Dr. Jennifer Elder

YMC 2024 | August 14th

Outline

(I) Background

(II) Classical Findings

(III) Homomesy & Subset Relations

(IV) Algebraic Perspective

Backgound

Homomesy lies within the field of *Dynamical Algebraic Combinatorics*, which can be thought of as the study of bijections on finite sets.

It is motivated by the work of James Propp, who originally thought of homomesy in the context of chip-firing games, it has since expanded to other areas.

Homomesy concerns three objects:

1. A finite set A.
2. A bijection $B : A \to A$.
3. A *statistic on* A, or a function $\mathcal{X} : A \to \mathbb{Z}$.

A triplet $(A, B, \mathcal{X})$ exhibits *homomesy* if the average value of a statistic $\mathcal{X}$ over each orbit $\mathcal{O}$ of a bijection B is constant, which is represented by:

$$\frac{1}{|\mathcal{O}|} \sum_{y \in \mathcal{O}} \mathcal{X}(y) = c.$$

In this case, we say $(A, B, \mathcal{X})$ is c-mesic.

In 1966, Konheim and Weiss published the paper where they presented a mathematical problem concerning the speed of storing and retrieval of records in a computer filing system.

Further ahead in the article, the same problem is presented in terms of n cars parking in n labelled spots on a one-way street.

Parking Functions | Alternative Definition

Let $[n] = \{1, .., n\}$. Let $f = (a_1, ...a_n) \in [n]^n$, and $f^{\uparrow} = (a_1^{\uparrow}, ..., a_n^{\uparrow})$ be the *non-decreasing rearrangement of* f. Then f is a parking function if $a_i^{\uparrow} \leq i$, for all $i \in [n]$.

$$f^{\uparrow} = (1, 2, 2, 2, 4, 6)$$

$$\uparrow$$

$$f = (2, 4, 2, 6, 1, 2)$$

Classical Findings

FindStat & SageMath

FindStat

Login

Navigation | All Collections | All Maps | All Statistics | Contribute Map | Contribute Statistic | Usage | Contributors | Citations | Interface

— database and search engine for combinatorial statistics and maps —

This **collaborative project** is

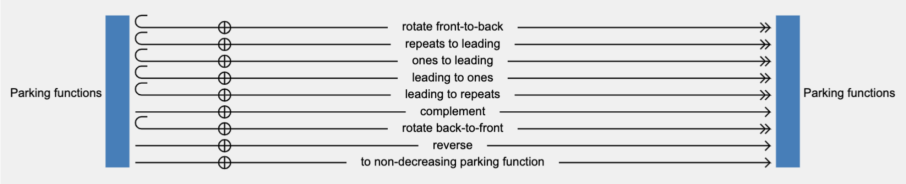
- a **database of combinatorial statistics and maps** on combinatorial collections and
- a **search engine**, identifying your data as the **composition of known maps and statistics**.
  - a **combinatorial collection** is a collection $\square = \cup_x \square_x$ of finite sets $\square_x$ (e.g. the set of permutations)
  - a **combinatorial map** is a map $\phi : \square \longrightarrow \square'$ between collections (e.g. the inverse of a permutation)
  - a **combinatorial statistic** (or **parameter**) is a map $\mathrm{st} : \square \longrightarrow \mathbb{Z}$ (e.g. the order of a permutation)
  - the database currently contains **1956 statistics** and **323 maps** on **24 collections**

There is a detailed usage example and several MathOverflow discussions with examples of the database usage.

You may also consult a discussion of the FindStat project that we compiled for **FPSAC 2019 conference**.

read more...

Parking Functions | Bijections & Statistics



Parking functions $\longrightarrow \mathbb{Z}$

St000135 The number of lucky cars of the parking function.
St000136 The dinv of a parking function.
St000165 The sum of the entries of a parking function.
St000188 The area of the Dyck path corresponding to a parking function and the total displacement of a parking function.
St000194 The number of primary dinversion pairs of a labelled dyck path corresponding to a parking function.
St000195 The number of secondary dinversion pairs of the dyck path corresponding to a parking function.
St000540 The sum of the entries of a parking function minus its length.
St000942 The number of critical left to right maxima of the parking functions.
St000943 The number of spots the most unlucky car had to go further in a parking function.
St001209 The pmaj statistic of a parking function.
St001903 The number of fixed points of a parking function.
St001904 The length of the initial strictly increasing segment of a parking function.
St001905 The number of preferred parking spots in a parking function less than the index of the car.
St001935 The number of ascents in a parking function.
St001937 The size of the center of a parking function.
St001946 The number of descents in a parking function.
St001947 The number of ties in a parking function.

Homomesy for Permutations

Currently, homomesy remains a new statistical concept with few papers in the field. Our study of homomesic bijections and statistics for parking functions has not been done before. However, permutations, which are a subset of parking functions, have already been studied:

- Elder, Jennifer, et al. "Homomesies on permutations: An analysis of maps and statistics in the FindStat database." *Mathematics of Computation* 93.346 (2024): 921-976

```
sage: from sage.databases.findstat import FindStatMaps, FindStatStatistics # Access to the FindStat methods
....: findstat()._allow_execution = True  # To run all the code from Findstat
....: for map in FindStatMaps(domain="Cc0001", codomain= "Cc0001"):  # Cc0001 is the Permutation Collection
....:     if map.properties_raw().find('bijective') >= 0:  # The map is bijective
....:         F = DiscreteDynamicalSystem(Permutations(n), map)   # Fix n ahead of time
....:         for stat in FindStatStatistics("Permutations"):
....:             if F.is_homomesic(stat):
....:                 print(map.id(), stat.id())
```

Sage Code w/ FindStat for Permutations

Only One Case?!

The rotate back to front map for parking functions is the bijection

$$B : (a_1, ..., a_n) \mapsto (a_n, a_1, ..., a_{n-1}).$$

Given a parking function f, the fixed point statistic $\mathcal{X}$ counts the number of fixed points of a parking function, or the values $i \in [n]$ where $a_i = i$.

Proposition: $(PF_n, B, \mathcal{X})$ is 1-mesic.

Homomesy and Subset Relations

Our instance of homomesy for parking functions can be generalized to $W_n = \{f : [n] \to [n]\}$, or "perfect words" of length n. Additionally, we know permutations exhibit the homomesy phenomenon, while parking functions don't.

New Goal: Investigate instances of homomesy induced by subset relationships

Instances of 0-Mesic Behavior

Sometimes, restricting a statistic to a specific subset of parking functions will result in the zero function.

- Cayley permutations have no *secondary dinversion pairs*.
- Prime parking functions have no *critical left-to-right maxima*.

Then these finite set and statistic pairs, with any bijection, will produce a case of 0-mesic behavior.

Trivial Homomesy for Permutations

Many parking function statistics are trivial over permutations, giving us easy cases of homomesy.

- Stat135, the lucky cars statistic, is constant.
- Stat943, the sum of entries, is also constant.
- Stat188, the area of the associated dyck path of a parking function, will always evaluate to 0 on permutations.

Nontrivial Homomesys

Proposition: Let SP_n be the set of Stirling permutations of length n, $B : SP_n \to SP_n$ be the reverse map, and $\mathcal{X} : SP_n \to \mathbb{Z}$ be the Dyck path area statistic. Then $(SP_n, B, \mathcal{X})$ is n^2-mesic.

Proposition: Let $B : S_n \to S_n$ be Map293, the rotate back to front map. Let $\mathcal{X}$ be Stat1905, which counts the number of preferences in a parking function less than the index of the car. Then $(S_n, B, \mathcal{X})$ is $\frac{n-1}{2}$-mesic.

Breather

There were many questions pervading our original research:

- Why is homomesy harder to find in parking functions?
- How can we meaningfully study the relationship between homomesy and subset relations?

Algebraic Perspective

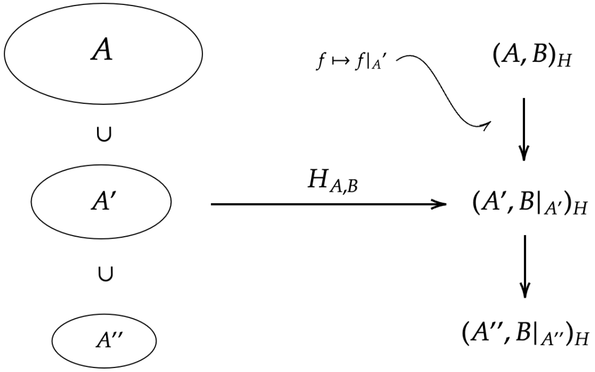
Naive Constructions

For a given combinatorial collection A and "graded" bijection $B : A \to A$, let $(A, B)_H$ be the $\mathbb{Z}$-module of all statistics X on A for which $(A, B, \mathcal{X})$ is homomesic.

If $A' \subset A$ is a combinatorial collection where $B|_{A'}$ is still a bijection, what is the relationship between $(A, B)_H$ and $(A', B|_{A'})_H$?

Interpreting Subset Relationships

$$A,\ B:\ A \to A$$



Formalization

We can define a presheaf $H_{A,B} : \mathcal{T}_{A,B}^{op} \to Mod_{\mathbb{Z}}$ by restriction, which represents how we think about homomesy.

Let $W = \bigcup_{n \in \mathbb{N}} \{f : [n] \to [n]\}$ be the collection of all "perfect words", $B : W \to W$ be the rotate back to front map, and $PF = \bigcup_{n \in \mathbb{N}} PF_n$, $S = \bigcup_{n \in \mathbb{N}} S_n$. Then $W \supset PF \supset S$. We can now consider $H_{W,B}(W \supset PF \supset S)$.

Initial Findings

Theorem: $H_{W,B}(W \supset PF)$ is surjective.

"Every instance of homomesy for PF and $B|_{PF}$ is the restriction of an instance of homomesy for W and B."

Proposition: Let $B :\to$ be the reverse map, and $\mathcal{X} : S \to S$ be Stat18, which counts the number of inversions of a permutation. Then $\mathcal{X} \in H_{,B}(S)$, but $\mathcal{X} \notin im(H_{PF,B}(PF \supset S))$.

Surjectivity

Parking functions and perfect words "have orbits of the same size"

$$g = (1,2,3,1,2,3) \in PF_6$$

$$O(g) \ = \ \{(1,2,3,1,2,3),\ (3,1,2,3,1,2),\ (2,3,1,2,3,1)\}$$

$$f = (6,5,6,6,5,6) \in W_6$$

$$O(f) \ = \ \{(6,5,6,6,5,6),\ (6,6,5,6,6,5),\ (5,6,6,5,6,6)\}$$

Sufficient?

This model can only classify instances of homomesy by orbit sizes, and doesn't account for the internal structures of the combinatorial collections. Why?

The Definitions

FindStat

Login

Navigation | All Collections | All Maps | All Statistics | Contribute Map | Contribute Statistic | Usage | Contributors | Citations | Interface

— database and search engine for combinatorial statistics and maps —

This **collaborative project** is

- a **database of combinatorial statistics and maps** on combinatorial collections and
- a **search engine**, identifying your data as the **composition of known maps and statistics**.
  - a **combinatorial collection** is a collection $\square = \cup_x \square_x$ of finite sets $\square_x$ (e.g. the set of permutations)
  - a **combinatorial map** is a map $\phi : \square \longrightarrow \square'$ between collections (e.g. the inverse of a permutation)
  - a **combinatorial statistic** (or **parameter**) is a map $\mathrm{st} : \square \longrightarrow \mathbb{Z}$ (e.g. the order of a permutation)
  - the database currently contains **1956 statistics** and **323 maps** on **24 collections**

There is a detailed usage example and several MathOverflow discussions with examples of the database usage.

You may also consult a discussion of the FindStat project that we compiled for **FPSAC 2019 conference**.

read more...

Summary of Results

- Discovered new instances of homomesy for parking functions and subsets of parking functions.
- Investigated new theory to study homomesy from a broader perspective.

Future Work

Classical Homomesy:

- Find more subsets of parking functions with nontrivial homomesy induced by restriction.
- Characterize all instances of homomesy for permutations induced by parking functions.

Algebra:

- Find stronger notions of a combinatorial collection
- Incorporate group actions

Thank You!

Any questions?

References

ppt

[1] Ashleigh Adams, Jennifer Elder, Nadia Lafrenière, Erin McNicholas, Jessica Striker, and Amanda Welch. The cyclic seiving phenonmenon on permutations, 2024. Manuscript is forthcoming.

[2] Sara C. Billey and Bridget E. Tenner. Fingerprint databases for theorems. *Notices Amer. Math. Soc.*, 60(8):1034–1039, 2013.

[3] S. Alex Bradt, Jennifer Elder, Pamela E. Harris, Gordon Rojas Kirby, Eva Reutercrona, Yuxuan (Susan) Wang, and Juliet Whidden. Unit interval parking functions and the r-fubini numbers. *La Mathematica*, pages 370–384, 2024.

[4] Jennifer Elder, Pamela E. Harris, Jan Kretschmann, and J. Carlos Martínez Mori. Parking functions, fubini rankngs, and boolean intervals in the weak order of $\mathfrak{S}_{,n}$. 2023. Accepted at Journal of Combinatorics.

[5] Jennifer Elder, Nadia Lafrenière, Erin McNicholas, Jessica Striker, and Amanda Welch. Homomesies on permutations: an analysis of maps and statistics in the findstat database. *Mathematics of Computation*, 2023.

[6] Martin Rubey, Christian Stump, et al. FindStat, - T,he combinatorial statistics database. http://www.FindStat.org,. Accessed: July 3, 2024.