Séminaire Lotharingien de Combinatoire **89B** (2023)
Article #91, 9 pp.

Proceedings of the 35th Conference on Formal Power Series and Algebraic Combinatorics (Davis)

A bijectionist's toolkit

Alexander Grosz*[1], Tobias Kietreiber†[2], Stephan Pfannerer‡[2], and Martin Rubey§[2]

[1]*Technical University of Munich, Germany*
[2]*Institute of Discrete Mathematics and Geometry, Technische Universität Wien, Austria*

Abstract. We present a tool, integrated into `SageMath` 10.0, which supports the combinatorialist in the discovery of an explicit bijection between two finite sets given various constraints, or by demonstrating that no such bijection can exist. As an application we present a conjecture which may have been hard to discover otherwise.

1 Introduction

Let $\mathcal{A} = (A_i)_{i\in I}$ and $\mathcal{B} = (B_i)_{i\in I}$ be two sequences of finite sets of the same size. One commonly employed approach for finding an explicit bijection between the two sets is to refine the problem by introducing new parameters or by stipulating that the bijection has other additional properties. For example, one may want to require that it intertwines natural actions or operations on $\mathcal{A}$ and $\mathcal{B}$ respectively or, provided $\mathcal{A} = \mathcal{B}$, that it is an involution. Also, one may want to find a parameter on $\mathcal{A}$ which has the same distribution as a natural parameter on $\mathcal{B}$ but satisfies additional constraints.

Thus, the first task is to discover such properties, and then determine experimentally whether the requirements can be met at all. To the best of the authors' knowledge, this is usually done by ad-hoc methods. For example, one might list the first few objects in $\mathcal{A}$ and $\mathcal{B}$ with pencil and paper or perhaps using a computer program and then inspect them manually.

While it is straightforward to check whether there are natural subsets of A_i and B_i that are still in bijection, other constraints may be very hard to verify or falsify. Moreover, databases like the `OEIS` [4] and `FindStat` [3] now enable us, in principle, to generate a

*alexander.grosz@tum.de. Alexander Grosz was supported by the Austrian Science Fund (FWF) P29275 and the Alexander von Humboldt Foundation with funds from the German Federal Ministry of Education and Research (BMBF).

†tobias.kietreiber@tuwien.ac.at. Tobias Kietreiber was supported by the Austrian Science Fund (FWF) P29275.

‡stephan.pfannerer@tuwien.ac.at. Stephan Pfannerer was supported by the Austrian Science Fund (FWF) P29275 and is a recipient of a DOC Fellowship of the Austrian Academy of Sciences.

§martin.rubey@tuwien.ac.at. Martin Rubey was supported by the Austrian Science Fund (FWF) P29275.

huge number of conjectural refinements, but it is infeasible to look into them by hand. The program we describe here seeks to simplify and automate this task.

The 'bijectionist's toolkit' is included in `SageMath` [5] beginning with version 10.0.[1]

2 Motivation

Before we present the features of the program in detail, let us describe the problem that originally motivated the development of the tool. Let $A = B = \mathfrak{S}_n$ and let $\operatorname{rot} : \mathfrak{S}_n \to \mathfrak{S}_n$ the 'rotation'-action on A, that is, conjugation with the long cycle $(1, \dots, n)$. Explicitly:

$$\operatorname{rot}(\pi)(i) = \begin{cases} \pi(i+1) - 1 & \text{if } \pi(i+1) \neq 1, \\ n & \text{otherwise,} \end{cases}$$

where we set $\pi(n+1) = \pi(1)$. Additionally, let $\tau : B \to \mathbb{N}$ be the length of a longest increasing subsequence in a permutation, that is, the largest number k such that $\pi(i_1) < \cdots < \pi(i_k)$ for some sequence $1 \leq i_1 < \cdots < i_k \leq n$.

Then, driven by a problem from invariant theory, we want to find a statistic on permutations, that is, a function $s : A \to \mathbb{N}$, which has the same distribution as τ but is additionally invariant under rot:

$$\begin{aligned} \sum_{\pi \in \mathfrak{S}_n} q^{s(\pi)} &= \sum_{\pi \in \mathfrak{S}_n} q^{\tau(\pi)}, \text{ and} \\ s(\operatorname{rot}(\pi)) &= s(\pi) \quad \text{for all } \pi \in \mathfrak{S}_n. \end{aligned}$$

Given these constraints the program is then able to produce a complete list of minimal subdistributions, that is, non-empty inclusion-wise minimal subsets $\tilde{A}$ of A together with subsets $\tilde{Z} \subset \mathbb{N}$ of the same cardinality, such that necessarily

$$\sum_{\pi \in \tilde{A}} q^{s(\pi)} = \sum_{z \in \tilde{Z}} q^z.$$

To illustrate, the program finds that the set of four permutations

$$[1, 2, 3, 4], [2, 3, 4, 1], [3, 4, 1, 2] \text{ and } [4, 1, 2, 3]$$

must be mapped by s to the four values $1, 2, 3$ and 4, in some order. A part of this is easy to explain: these are exactly the four permutations which are invariant under rotation, so all other orbits of rot have cardinality at least 2. However, there is only a single permutation with length of longest increasing subsequence equal to 1 and a single

[1]The full documentation is available at https://doc.sagemath.org/html/en/reference/combinat/sage/combinat/bijectionist.html

permutation with the length of a longest increasing subsequence equal to 4. Thus, these two values can only be images of permutations which are invariant under rotation.

Somewhat to our disappointment, even with the help of our tool we were unable to find an explicit description of the statistic s. Still, possibly encouraged by the fact that the program demonstrated that the statistic exists for $n \leq 9$, the existence eventually could be proven [1].

Note that the existence of such a statistic is equivalent to the existence of a bijection $S : \mathfrak{S}_n \to \mathfrak{S}_n$ such that $(\tau \circ S)(\operatorname{rot} \pi) = (\tau \circ S)(\pi)$.

3 Features

Let us now describe the main features of the tool in full generality. All types of constraints can be combined.

Basic setup

Let A and B be finite sets of equal size, and let $\tau : B \to Z$ for an arbitrary set Z. Then

```
sage: bij = Bijectionist(A, B, tau)
```

initializes `bij` to represent all functions $s : A \to Z$ such that there is a bijection $S : A \to B$, such that the following diagram commutes:

$$\begin{tikzcd} A \arrow[d, "S"'] \arrow[dr, "s"] & \\ B \arrow[r, "\tau"'] & Z \end{tikzcd}$$

As a special case we allow τ to be the identity map on B. In this case `bij` represents all bijections $S : A \to B$, and we must have $Z = B$ and $S = s$.

Constraints

Statistics

We may supply a pair (or several pairs) of statistics $\alpha : A \to W$ and $\beta : B \to W$ on A and B respectively into an arbitrary set W:

```
sage: bij.set_statistics((alpha, beta))
```

Doing so, we stipulate that the following diagram commutes:

$$\begin{tikzcd} A \arrow[d, "S"'] \arrow[dr, "\alpha"] & \\ B \arrow[r, "\beta"'] & W \end{tikzcd}$$

In words, we require that S is an intertwining bijection between the statistics α and β.

Intertwining relations

We may supply a pair (or several pairs) of k-ary maps $\pi : A^k \to A$ and $\rho : Z^k \to Z$, thus requiring that they are intertwined by s:

```
sage: bij.set_intertwining_relations((k, pi, rho))
```

Put differently, the following diagram must then commute:

$$\begin{tikzcd} A^k \arrow[r, "\pi"] \arrow[d, "s^k"'] & A \arrow[d, "s"] \\ Z^k \arrow[r, "\rho"'] & Z \end{tikzcd}$$

One may want to think of π as a map which composes objects in A, and ρ as the effect this should have on the statistic we are looking for.

Quadratic relations

We may supply a pair (or several pairs) of maps $\phi : A \to Z$ and $\psi : Z \to A$,

```
sage: bij.set_quadratic_relations((phi, psi))
```

thus requiring that the following diagram commutes:

$$\begin{tikzcd} A \arrow[r, "s"] \arrow[d, "\phi"'] & Z \arrow[d, "\psi"] \\ Z & A \arrow[l, "s"] \end{tikzcd}$$

In particular, if ϕ and ψ are the identity maps (and therefore $Z = A = B$ and $S = s$), this requires that S is an involution on A.

Constant blocks

For a set partition P of A, setting

```
sage: bij.set_constant_blocks(P)
```

we stipulate that $s : A \to Z$ is constant on the blocks of P. Note that this makes sense only if τ is not injective. To illustrate, this accommodates the condition of the motivating example that the statistic should be invariant under rot, by letting P be the corresponding orbit decomposition.

Homomesy constraints

For a set partition Q of A, setting

```
sage: bij.set_homomesic(Q)
```

we assert that the average of s on each block of Q should be constant. Of course, this requires that Z is a field.

Solutions

The three most important ways to explore the solution space are as follows.

Fibers of the statistics

The instruction

```
sage: bij.statistics_fibers()
```

simply returns the preimages of $\alpha : A \to W$ and $\beta : B \to W$. Note that it does not assert that a statistic s exists.

All solutions

Issuing

```
sage: list(bij.solutions_iterator())
```

the program lists all functions $s : A \to Z$ which satisfy all the given constraints, or raises an error if no such map exists. Frequently, however, the number of such functions will be very large. Thus, and in particular if one is only interested in checking whether the constraints can be satisfied at all, it is more useful to use `next(bij.solutions_iterator())`, which only outputs a single solution, if it exists. In any case, this is computationally expensive.

Minimal subdistributions

```
sage: list(bij.minimal_subdistributions_iterator())
```

lists all minimal subdistribution implied by the given constraints, or raises an error if no solution exists. More precisely, it returns all non-empty inclusion-wise minimal subsets $\tilde{A}$ of A of together with submultisets $\tilde{Z} \subset Z$ of the same cardinality, such that we have $s(\tilde{A}) = \tilde{Z}$ as multisets for all possible solutions. In particular, if τ is the identity map on B, then we obtain a pair of subsets of minimal cardinality of A and B respectively which are in bijection for all maps s satisfying the given constraints. This is computationally expensive.

4 A sample application

A treasure trove for the experimental combinatorialist is the database `FindStat`. It should not come as a surprise that combining the toolkit with `FindStat` yields some potentially interesting conjectures.

Recently, Elder, Lafrenière, McNicholas, Striker and Welch [2] used the database to find homomesies on permutations, that is, statistics $s : \mathfrak{S}_n \to \mathbb{Z}$ and bijections $m : \mathfrak{S}_n \to \mathfrak{S}_n$ such that there exists a constant $c \in \mathbb{Q}$ with

$$\frac{1}{|\mathcal{O}|} \sum_{\pi \in \mathcal{O}} s(\pi) = c$$

for all orbits $\mathcal{O}$ of m. Using a brute force search, they discovered more than a hundred combinations of maps and conjecturally homomesic statistics. Remarkably, they were then able to prove that all of these conjectures are actually true.

Using our toolkit, we can take this a step further and look for statistics τ which are *equidistributed* to a homomesic statistic, given a map m:

```
sage: n = 5
sage: A = B = Permutations(n)
sage: Q = DiscreteDynamicalSystem(A, m).cycles()
sage: bij = Bijectionist(A, B, tau)
sage: bij.set_homomesic(Q)
sage: next(bij.solutions_iterator())
```

If this yields a solution, we can be quite confident that s exists. For example, if m is the rotation action rot considered before, iterating through the statistics in `FindStat` yields www.findstat.org/St001377, 'The major index minus the number of inversions of a permutation' as a candidate.

As a next step, we can use `FindStat` again to refine our conjecture, by iterating over all pairs of equidistributed statistics α and β, such that

```
sage: bij.set_statistics((alpha, beta))
sage: next(bij.solutions_iterator())
```

still gives a solution. In the case at hand, there are very many such pairs. One particularly natural refinement is to use the major index for both α and β. We now look at the minimal subdistributions in the refined setting, for $n = 4$:

```
sage: list(bij.minimal_subdistributions_iterator())
 [([[4, 3, 2, 1]], [0]),
  ([[4, 1, 2, 3]], [0]),
  ([[3, 4, 1, 2]], [0]),
  ([[2, 3, 4, 1]], [0]),
  ([[2, 1, 4, 3]], [0]),
  ([[1, 2, 3, 4]], [0]),
  ([[2, 1, 3, 4], [3, 1, 2, 4]], [-2, -1]),
 ...
```

We can, in fact, query `FindStat` again with this data:

```
sage: findstat(list(bij.minimal_subdistributions_iterator()))
 0: St001377oMp00064oMp00073 (quality [100, 100])
 ...
```

Although this is a plausible result, it turns out that this statistic is not homomesic with respect to rotation. However, we can make the following conjecture, which we tested up to and including $n = 6$.

Conjecture 1. *Let* $\mathrm{maj} : \mathfrak{S}_n \to \mathbb{N}$ *be the major index and let* $\mathrm{inv} : \mathfrak{S}_n \to \mathbb{N}$ *be the number of inversions of a permutation. Furthermore, let* $\mathrm{rot} : \mathfrak{S}_n \to \mathfrak{S}_n$ *be the conjugation of a permutation with the long cycle* $(1, \ldots, n)$*.*

There is a statistic $s : \mathfrak{S}_n \to \mathbb{Z}$ *which is homomesic with respect to* rot*, and such that the pairs of statistics* (s, maj) *and* $(\mathrm{maj} - \mathrm{inv}, \mathrm{maj})$ *are equidistributed, that is,*

$$\sum_{pi \in \mathfrak{S}_n} q^{s(\pi)} t^{\mathrm{maj}(\pi)} = \sum_{pi \in \mathfrak{S}_n} q^{\mathrm{maj}(\pi) - \mathrm{inv}(\pi)} t^{\mathrm{maj}(\pi)}.$$

5 Implementation details

In this section, we outline the basic ideas behind the implementation. We model the problem as a 0-1 integer linear program that represents the statistic s. Given a (possibly trivial) set partition P of A and given Z, we define binary variables $x_{p,z}$ for each $p \in P$ and $z \in Z$. The assignment $x_{p,z} = 1$ then indicates that $\forall a \in p : s(a) = z$, thereby encoding that s is constant on the blocks of P.

To ensure that s and τ have the same distribution, and that the bijection $S : A \to B$ realising this equidistribution respects the statistics $\alpha : A \to W$ and $\beta : B \to W$ we add the following constraints:

$$\sum_{p \in P} m_w(p) x_{p,z} = n_w(z), \qquad \forall w \in W \ \forall z \in Z$$

$$\sum_{z \in Z} x_{p,z} = 1, \qquad \forall p \in P$$

where $m_w(p) = |\{a \in p : \alpha(a) = w\}|$ and $n_w(z) = |\{b \in B : \beta(b) = w \text{ and } \tau(b) = z\}|$.

A solution to this integer program immediately yields a solution for $s : A \to Z$. To generate all possible solutions for s, we iteratively prohibit the current solution by adding the linear constraint $\sum_{p \in P} x_{p,s(p)} < |P|$, where $s(p)$ denotes the common image of all $a \in p$.

The minimal subdistributions can be algorithmically determined by means of a dynamic optimization problem that interacts with the model above. In every iteration, we guess a smallest subset of blocks p that potentially defines a minimal subdistribution. By an auxiliary integer program we find a minimal selection of blocks with compatible distributions according to all the solutions of the base problem we have encountered so far. We then attempt to confirm the conjecture that the incumbent subdistribution holds for any solution by testing the feasibility of the base problem after prohibiting the specific guess at hand. In case of infeasibility, we terminate with the given guess; otherwise we add the information of the new solution to create a new guess for a minimal selection of blocks. Analogously, we can determine an improved partition P' of A that is implicitly forced by other restrictions of the statistic.

Acknowledgements

We want to thank Matthias Köppe and Travis Scrimshaw for reviewing the code and thus allowing integration into the upcoming version 10.0 of `SageMath`.

References

[1] P. Alexandersson, S. Pfannerer, M. Rubey, and J. Uhlin. “Skew characters and cyclic sieving”. *Forum Math. Sigma* **9** (2021), Paper No. e41, 31. DOI.

[2] J. Elder, N. Lafrenière, E. McNicholas, J. Striker, and A. Welch. “Homomesies on permutations – an analysis of maps and statistics in the `FindStat` database”. 2022. arXiv:2206.13409.

[3] M. Rubey, C. Stump, et al. “FindStat - The combinatorial statistics database”. Link.

[4] N. J. A. Sloane and T. O. F. Inc. “The on-line encyclopedia of integer sequences”. 2023. Link.

[5] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 10.0.beta4)*. 2023. Link.