

# MATH502 — Combinatorics 2

Based on the lectures by Maria Gillespie

Notes written by Ignacio Rojas

Spring 2023

Please note that these notes were not provided or endorsed by the lecturer and have been significantly altered after the class. They may not accurately reflect the content covered in class and any errors are solely my responsibility.

This is the second semester of an introductory graduate-level course on combinatorics. We will be covering symmetric function theory, Young tableaux, counting with group actions, designs, matroids, finite geometries, and not-so-finite geometries.

The goal of this class is to give an overview of the wide variety of topics and techniques in both classical and modern combinatorial theory.

## Requirements

Knowledge on theory of enumeration, generating functions, combinatorial species, the basics of graph theory, posets, partitions and tableaux, and basic symmetric function theory is required.

# Contents

<b>Contents</b>	<b>2</b>
<b>1 Symmetric functions</b>	<b>5</b>
1.1 Day 1   20230120	5
1.2 day 2	8
1.3 Day 3   20230125	11
1.4 Day 4   20230127	14
1.5 Interim 1	15
1.6 Day 5   20230130	16
1.7 Day 6   20230201	19
1.8 Day 7   20230203	22
1.9 Day 8   20230206	25
1.10 Day 9   20230208	28
1.11 Day 10   20230210	29
1.12 Day 11   20230213	33
1.13 Day 12   20230215	36
1.14 Day 13   20230217	37
1.15 Day 14   20230220	41
<b>2 Representation Theory: A Crash Course</b>	<b>45</b>
2.1 Day 15   20230222	45
2.2 Day 16   20230224	48
2.3 Day 17   20230227	51
2.4 Day 18   20230301	53
2.5 Day 19   20230303	55
2.6 Day 20   20230306	58
<b>3 Counting with Group Actions</b>	<b>63</b>

3.1	Day 21   20230308	63
3.2	Day 22   20230310	66
<b>Index</b>		<b>69</b>
<b>Bibliography</b>		<b>71</b>



# Chapter 1

## Symmetric functions

### 1.1 Day 1 | 20230120

**Definition 1.1.1.**  $f(x_1, x_2, \dots)$  is symmetric if it's fixed under permutations of variables. For a permutation  $\sigma$  this is,

$$f(x_{\sigma(1)}, x_{\sigma(2)}, \dots) = f(x_1, x_2, \dots).$$

**Example 1.1.2.** The function

$$f(x_1, \dots, x_4) = x_1^5 + \dots + x_4^5$$

is known as  $p_5$  or  $m_{(5)}$ , where  $p$  is the power-sum symmetric function and  $m$ , the monomial symmetric function.

We can have the function defined on infinitely many variables. Consider the function  $g$  defined as

$$g = x_1^4 x_2 + x_1^4 x_3 + \dots + x_i^4 x_j + \dots + 3x_1 + \dots + 3x_i + \dots = m_{(4,1)} + 3m_{(1)}.$$

Let us recall some **notation**,

$$\begin{cases} \Lambda_R(x_1, \dots, x_n) \rightarrow \text{symmetric functions on } n \text{ variables over } R, \\ \Lambda_R(\underline{x}) \rightarrow \text{symmetric functions on infinitely many variables over } R. \end{cases}$$

In our case  $R = \mathbb{Q}$ , so the object of study is  $\Lambda_{\mathbb{Q}}$ .

**Proposition 1.1.3.** *The space  $\Lambda_{\mathbb{Q}}^n$  is the space of symmetric functions of degree  $n$ . Its dimension is  $p(n)$ , the number of partitions of  $n$ .*

This is because, for every such function we can decompose it into monomials and the monomial symmetric functions form a basis.

## 1. SYMMETRIC FUNCTIONS

---

### Bases of $\Lambda_Q$

Suppose  $\lambda = (\lambda_1, \dots, \lambda_k) \vdash n$  with  $\lambda_1 \geq \dots \geq \lambda_k$ .

### Monomial Symmetric Functions

The function  $m_\lambda(\underline{x})$  is the smallest symmetric function which contains the monomial  $x_1^{\lambda_1} x_2^{\lambda_2} \dots x_k^{\lambda_k}$  as a term. In general

$$m_\lambda = \sum_{i_1 \neq \dots \neq i_k} x_{i_1}^{\lambda_1} \dots x_{i_k}^{\lambda_k}.$$

**Example 1.1.4.** Consider the partition  $(5, 3) \vdash 8$ . The function  $m_{(5,3)}$  will be different depending on the number of variables:

- ◇ In one variable we can't have monomials of the form  $x_i x_j$ , so  $m_{(5,3)} = 0$ .
- ◇ In two variables we have  $m_{(5,3)}(x, y) = x^5 y^3 + y^5 x^3$ .
- ◇ In three variables the function is

$$m_{(5,3)}(x, y, z) = x^5 y^3 + y^5 z^3 + z^5 x^3 + y^5 x^3 + z^5 y^3 + x^5 z^3.$$

Considering some special cases, take the partition  $(1, 1, 1, 1) \vdash 4$ , then

$$\begin{aligned} m_{(1,1,1,1)}(u, v, x, y, z) &= uvxy + vxyz + xyzu + yzuv + zuvx \\ &= uvxy + uxyz + uvyz + uvxz + vxyz. \end{aligned}$$

For cases with less than 4 variables the function is zero and in exactly four, it has 1 term. The partition  $(4) \vdash 4$  returns the function

$$m_{(4)}(x) = x^4, \quad m_{(4)}(x, y) = x^4 + y^4, \quad m_{(4)}(x, y, z) = x^4 + y^4 + z^4,$$

and so on with any number of variables.

*Remark 1.1.5.* The number of terms in  $m_\lambda(x_1, \dots, x_d)$  is **I actually don't know**, while the degree of  $m_\lambda$  is  $|\lambda| = n$ .

### Elementary Symmetric Functions

**Definition 1.1.6.** For any  $r \in \mathbb{N}$ , the elementary symmetric function  $e_r$  is  $m_{(1,1,\dots,1)}$  ( $r$  ones). For  $\lambda$ , a partition,  $e_\lambda = \prod e_{\lambda_i}$ . As an alternative for  $m_{(1,1,\dots,1)}$  we can also write

$$e_r(x_1, \dots, x_d) = \sum_{1 \leq i_1 < \dots < i_r \leq n} x_{i_1} \dots x_{i_r}.$$

**Example 1.1.7.** Let us calculate  $e_{(2,1)}$  for 1 through 3 variables. When we have  $e_{(2,1)}(x) = e_2(x)e_1(x)$ , we can't compute  $e_2(x)$  because there are no two-term monomials with only one variable. On two variables we have the following

$$e_{(2,1)}(x, y) = e_2(x, y)e_1(x, y) = (xy)(x + y) = x^2y + y^2x$$

and when talking about 3 variables the following happens:

$$\begin{aligned} e_{(2,1)}(x, y, z) &= e_2(x, y, z)e_1(x, y, z) \\ &= (xy + yz + zx)(x + y + z) \\ &= x^2y + y^2z + z^2x + y^2x + z^2y + x^2z + 2xyz. \end{aligned}$$

Consider now the partitions  $(2, 2, 2, 2)$  and  $(5)$ . Then

$$e_{(2,2,2,2)} = e_2^4 \Rightarrow e_{(r,r,\dots,r)} = e_r^{m_r(\lambda)}$$

where  $m_i(\lambda)$  is number of parts of  $\lambda$  equal to  $i$ . For the partition  $(5)$  we have that  $e_{(5)} = e_5$  and in general  $e_{(n)} = e_n$ .

*Remark 1.1.8.* As before **we don't know how many terms per function**, but knowing  $m$  implies knowing  $e$ . As for the degree, it holds that  $\deg(e_\lambda) = |\lambda|$ .

## Homogenous Symmetric Functions

◇ Homogenous:  $h_\lambda = \prod h_{\lambda_i}$  and  $h_d = x_1^d + \dots + x_1^{d-1}x_2 + \dots + x_1^{d-2}x_2^2 + x_1^{d-2}x_2x_3 + \dots$   
In general  $h_d = \sum_{\lambda \vdash d} m_\lambda$ .

◇ Power sum:  $p_\lambda = \prod p_{\lambda_i}$  and  $p_d = \sum x_i^d$ .

For Schur basis recall SSYT

**Example 1.1.9.** Consider  $\lambda = (5, 4, 1)$ , rows  $\leq \rightarrow$  and columns  $<$ , we associate the monomial  $x_1^2x_2^3x_3^3x_4^2 := x^T$ .

◇ Schur:  $s_\lambda = \sum_{T \in SSYT(\lambda)} x^T$  but also  $\sum K_{\lambda\mu} m_\mu$  where the sum is over SSYT of shape  $\lambda$ , content  $\mu$ .

### Schur function motivation (preview)

The first place they showed up is in the representation theory of Lie group. The function  $s_\lambda(x_1, \dots, x_n)$  is a character of irreducible polynomial representations of  $GL_n$ . In theoretical physics we have matrix groups acting on particles, representations are smaller matrix groups of things that they are mapping to. We want to take tensor product and direct sums of representations, the tensor product is related to multiplication of Schur function while direct sum into sum of Schur functions.

There's also the Schur-Weyl duality which takes representations into the Weyl group. Under the *Frobenius map*,  $s_\lambda$  corresponds to irreducible representations of  $S_n$ .

A more modern application of Schur function goes into geometry,  $s_\lambda$  correspond to Schubert varieties in Grassmannians. Multiplication corresponds to interesections and sum to unions.

There's also context in Probability Theory. But in the end, Schur positivity is important because of this connections.

**Definition 1.1.10.**  $f \in \Lambda$  is Schur-positive if  $f = \sum c_\lambda s_\lambda$ ,  $c_\lambda \geq 0$ .

**Example 1.1.11.**  $3s_{(2,1)} + 2s_{(3)}$  schur pos but change 2 to  $-\frac{1}{2}$  then not.

## 1.2 day 2

### Alg defn Schur fncs

**Definition 1.2.1.** A function is antisymmetric if for  $\pi \in S_n$ ,

$$f(x_{\pi(1)}, \dots, x_{\pi(n)}) = \text{sgn}(\pi) f(x_1, \dots, x_n).$$

**Example 1.2.2.** The following functions are antisymmetric:

- (a)  $f(x, y) = x - y$  then  $f(y, x) = -f(x, y)$ .
- (b)  $g(x, y) = (x - y)(x + y)$ .
- (c)  $h(x, y) = x^2y - y^2x$ .

Notice that the last function can factor as  $h = -xy(x - y)$ . We claim that this is always the case.

**Lemma 1.2.3.** Every antisymmetric polynomial  $f$  in two variables  $x, y$  can factor as  $f(x, y) = (x - y)g(x, y)$  where  $g$  is symmetric.



**Proof**

Suppose  $f$  is antisymmetric, then  $f(x, x) = 0$  by taking  $y = x$ . This means that  $(x - y) \mid f$ . Thus  $f(x, y) = (x - y)g(x, y)$  and we now need to show that  $g$  is symmetric.

$$g(y, x) = \frac{f(y, x)}{y - x} = \frac{-f(x, y)}{-(x - y)} = \frac{f(x, y)}{x - y} = g(x, y).$$

**Monomial Antisymmetric Functions**

**Definition 1.2.4.** Given a strict partition  $\lambda = (\lambda_1, \dots, \lambda_k)$ ,  $\lambda_1 > \dots > \lambda_k$ , we define

$$a_\lambda(x_1, \dots, x_n) = x_1^{\lambda_1} \cdots x_k^{\lambda_k} \pm \text{similar terms} = \sum_{\pi \in S_n} \text{sgn}(\pi) \prod_k x_{\pi(k)}^{\lambda_k}.$$

This  $a_\lambda$  can be zero.

**Example 1.2.5.** For two variables we've seen some antisymmetric polynomials. Let us calculate

$$a_{(3,1)}(x, y) = x^3y - y^3x.$$

The smallest possible example in 3 variables is

$$a_{(2,1,0)}(x, y, z) = x^2y + y^2z + z^2x - y^2x - z^2y - x^2z.$$

This can be factored as  $(x - y)(y - z)(x - z)$ . A similar construction gives us

$$a_{(4,2,0)}(x, y, z) = x^4y^2 + y^4z^2 + z^4x^2 - y^4x^2 - z^4y^2 - x^4z^2,$$

but how does this factor? We get

$$a_{(4,2,0)}(x, y, z) = (x^2 - y^2)(y^2 - z^2)(x^2 - z^2) = a_{(2,1,0)}(x, y, z)(x + y)(y + z)(x + z).$$

**Lemma 1.2.6.** The set  $\{a_\lambda\}_{\lambda \text{ strict}}$  is a basis of the antisymmetric polynomials over  $\mathbb{Q}$ ,  $A_{\mathbb{Q}}$ . Even more any  $a_\lambda$  is divisible by  $a_\rho$  where  $\rho = (n - 1, n - 2, \dots, 2, 1, 0)$ .

As an algebra generator,  $a_\rho$  is a generator.

**Proof**

WRITE

## 1. SYMMETRIC FUNCTIONS

---

**Proposition 1.2.7.** *The  $a_\rho$  antisymmetric function is also the Vandermonde determinant:*

$$a_\rho = \det \begin{pmatrix} x_1^{n-1} & x_1^{n-2} & \dots & x_1^2 & x_1 & 1 \\ x_2^{n-1} & x_2^{n-2} & \dots & x_2^2 & x_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_n^{n-1} & x_n^{n-2} & \dots & x_n^2 & x_n & 1 \end{pmatrix}$$

### Schur Polynomials

**Definition 1.2.8.** The Schur polynomial of  $\lambda \in \text{Par}$  is

$$s_\lambda(x_1, \dots, x_n) = \frac{a_{\lambda+\rho}(\underline{x})}{a_\rho(\underline{x})}.$$

Here  $\lambda + \rho$  is the pointwise sum as arrays.

*Remark 1.2.9.* This is the Weyl character proof.

The following proof is due to Proctor(1987) [find ref](#)

**Lemma 1.2.10.** *Any  $a_\lambda$  can be seen as a determinant in the following way:*

$$a_\lambda(\underline{x}) = \det \begin{pmatrix} x_1^{\lambda_1} & x_1^{\lambda_2} & \dots & x_1^{\lambda_n} \\ x_2^{\lambda_1} & x_2^{\lambda_2} & \dots & x_2^{\lambda_n} \\ \vdots & \vdots & \ddots & \vdots \\ x_n^{\lambda_1} & x_n^{\lambda_2} & \dots & x_n^{\lambda_n} \end{pmatrix}$$

#### Proof

We want to see that

$$\frac{a_{\lambda+\rho}(\underline{x})}{a_\rho(\underline{x})} = \sum x^T$$

where the sum ranges through  $T$ 's which are SSYT(la) with max entry  $n$ .

- (a) We will show a recursion for the combinatorial definition that the character formula will also satisfy. It holds that

$$s_\lambda(\underline{x}) = \sum s_\mu(\underline{x}) x_n^{|\lambda| - |\mu|}$$

where  $\mu$  has  $n - 1$  parts with  $\lambda_1 \geq \mu_1 \geq \lambda_2 \geq \mu_2 \dots$

- (b) We also show that the ratio of determinants satisfies the same recursion.

**Example 1.2.11.** Consider  $\lambda = (8, 8, 4, 1, 1)$  and  $\mu = (8, 5, 2, 1)$ , then  $\lambda \setminus \mu$  is a skew-table in which we can fill in  $n$ 's

**Corollary 1.2.12.** *The Schur polynomials are a basis of  $\Lambda_{\mathbb{Q}}$ .*

### 1.3 Day 3 | 20230125

Recall  $\Lambda = \mathbb{Q}[e_1, e_2, \dots]$  where the  $e_j$ 's are the elementary symmetric functions. So the  $e_j$ 's are algebraic generators of  $\Lambda$  and they're algebraically independent. Equivalently, as a vector space,  $\{e_{\lambda} : \lambda \in \text{Par}\}$  is a basis.

**Proposition 1.3.1.** *A homomorphism  $f : \Lambda \rightarrow \Lambda$  ( $f(a+b) = f(a) + f(b)$ ,  $f(ab) = f(a)f(b)$  for  $a, b \in \Lambda$ ) is fully determined by where it sends the  $e_i$ 's.*

**Definition 1.3.2.** The map  $\omega \in \text{End}(\Lambda)$  will send  $e_j$  to  $h_j$ .

**Example 1.3.3.** Consider  $f = 3e_{(2,1)} + 2e_3$ , then applying  $\omega$  we get

$$\omega(f) = \omega(3e_{(2,1)} + 2e_3) = 3h_{(2,1)} + 2h_3.$$

For  $p_2$ , we can decompose to  $e_1^2 - 2e_2$ . So

$$\omega(p_2) = \omega(e_1^2 - 2e_2) = h_1^2 - 2h_2$$

and we can expand this last expression into

$$(x_1 + x_2 + \dots)^2 - 2(x_1^2 + x_2^2 + \dots + x_1x_2 + x_1x_3 + \dots) = -x_1^2 - x_2^2 - \dots$$

and we recognize this last term as  $-p_2$ . *This is not a coincidence.*

**Theorem 1.3.4.** *The map  $\omega$  is involutive.*

#### Proof

It suffices to prove that  $\omega(h_j) = e_j$ . We will use power expansions and generating functions. We have

$$H(t) = \frac{1}{1 - x_1 t} \frac{1}{1 - x_2 t} \cdots = \sum h_n(\underline{x}) t^n,$$

and this comes from expanding the  $1/(1 - y)$ 's as geometric series. When collecting the coefficients of  $t^n$  we get exactly  $h_n(\underline{x})$ . Similarly, for the elementary

symmetric functions,

$$E(t) = (1 + x_1 t)(1 + x_2 t) \cdots = \sum e_n t^n.$$

When multiplying to obtain the coefficient of  $t^n$  we get a plethora of different  $x_j$ 's which form the  $e_j$ 's. Now from this expressions we have  $H(t)E(-t) = 1$  which means that

$$\left( \sum h_n(\underline{x}) t^n \right) \left( \sum e_n(\underline{x}) (-t)^n \right) \Rightarrow \sum_{k=0}^n (-1)^k e_k h_{n-k} = 0, \quad n \geq 1.$$

Now applying the map to the equation we get

$$\omega \left( \sum_{k=0}^n (-1)^k e_k h_{n-k} \right) = \sum_{k=0}^n (-1)^k h_k \omega(h_{n-k}) = 0.$$

After reindexing, we get that both  $e_j$ 's and  $\omega(h_j)$ 's are determined recursively by the  $h_j$ 's in the same way. Thus we conclude that  $\omega(h_j) = e_j$ .

**Lemma 1.3.5.** *The following equation holds for the power-sum symmetric functions:*

$$\exp \left( \sum \frac{1}{n} p_n(\underline{x}) p_n(\underline{y}) \right) = \prod_{i,j=1}^{\infty} \frac{1}{1 - x_i y_j} = : \Omega(\underline{x}, \underline{y}).$$

It also holds that

$$\Omega(\underline{x}, \underline{y}) = \sum_{\lambda} a_{z_{\lambda}} \frac{1}{z_{\lambda}} p_{\lambda}(\underline{x}) p_{\lambda}(\underline{y})$$

where  $z_{\lambda} = \prod k^{m_k} m_k!$  where  $m_k$  is the number of parts of  $\lambda$  equal to  $k$ .

#### Proof

We will prove both parts separately. For the first equation we will take the logarithm on both sides:

$$\sum \frac{1}{n} p_n(\underline{x}) p_n(\underline{y}) = \log \left( \prod_{i,j=1}^{\infty} \frac{1}{1 - x_i y_j} \right)$$

and after manipulating the logarithm we get

$$\sum_{i,j=1}^{\infty} (\log(1) - \log(1 - x_i y_j)) = \sum_{i,j=1}^{\infty} \sum_{n=1}^{\infty} \frac{1}{n} x_i^n y_j^n.$$

We can separate<sup>a</sup> into

$$\sum_{n=1}^{\infty} \frac{1}{n} \left( \sum_i x_i^n \right) \left( \sum_j y_j^n \right).$$

Now taking exp on both sides we get equality.

By not removing the exponential we get the following expression

$$\exp \left( \sum_n \frac{1}{n} p_n(\underline{x}) p_n(\underline{y}) \right) = \sum_{k=0}^{\infty} \frac{1}{k!} \left( \sum_n \frac{1}{n} p_n(\underline{x}) p_n(\underline{y}) \right)^k.$$

To get a term of the form  $p_\lambda(\underline{x}) p_\lambda(\underline{y})$  we have to choose which parts of the  $\lambda$  come from each of the factors in  $\sum_n \frac{1}{n} p_n(\underline{x}) p_n(\underline{y})$ . If  $\ell(\lambda) = k$  then it comes from the  $k^{\text{th}}$  term in the exponential sum. If  $\lambda = (\lambda_1, \text{dots}, \lambda_1, \dots, 2, \dots, 2, 1, \dots, 1)$  with  $m_{\lambda_1}$   $\lambda_1$ 's,  $m_1$  1's, then out of  $k$  elements we have to choose  $m_1$  1's and so on. Thus there are  $\binom{k}{m_{\lambda_1}, \dots, m_1}$  choices and each  $i$  in  $\lambda$  comes with a  $\frac{1}{i}$ . Therefore the coefficient of  $p_\lambda(\underline{x}) p_\lambda(\underline{y})$  is

$$\frac{1}{k!} \frac{k!}{m_1! m_2! \dots} \frac{1}{1^{m_1}} \frac{1}{2^{m_2}} \dots = \frac{1}{z_\lambda}.$$

<sup>a</sup>Are we using Fubini-Tonelli here?

**Lemma 1.3.6.** *We have the following identities*

$$\exp \left( \sum_n \frac{(-1)^{n-1}}{n} p_n(\underline{x}) p_n(\underline{y}) \right) = \prod_{i,j=1}^{\infty} \frac{1}{1 + x_i y_j} = \sum_{\lambda} \frac{(-1)^{n-\ell(\lambda)}}{z_\lambda} p_\lambda(\underline{x}) p_\lambda(\underline{y}).$$

**Lemma 1.3.7.** *Another equality for  $\Omega(\underline{x}, \underline{y})$  is*

$$\Omega(\underline{x}, \underline{y}) = \sum_{\lambda} m_{\lambda}(\underline{x}) h_{\lambda}(\underline{y})$$

**Theorem 1.3.8.** *It holds that  $\omega(p_\lambda) = (-1)^{n-k} p_\lambda$  where  $k$  is the number of parts of  $\lambda$ .*

#### Proof

Applying  $\omega$  to  $\Omega$ , but only working with  $\underline{y}$  variables we get

$$\omega(\Omega) = \omega \left( \sum_{\lambda} m_{\lambda}(\underline{x}) h_{\lambda}(\underline{y}) \right) = \sum_{\lambda} m_{\lambda}(\underline{x}) e_{\lambda}(\underline{y}) = \prod_{i,j=1}^{\infty} (1 + x_i y_j) = \sum_{\lambda} \frac{1}{z_{\lambda}} (-1)^{n-k_{\lambda}} p_{\lambda}(\underline{x}) p_{\lambda}(\underline{y}).$$

## 1. SYMMETRIC FUNCTIONS

---

Comparing coefficients with

$$\omega \left( \sum_l a \frac{1}{z^\lambda} p_\lambda(\underline{x}) p_\lambda(\underline{y}) \right)$$

we get the result.

### 1.4 Day 4 | 20230127

To continue exploring the ring of symmetric functions we need a couple of tools. One of them is the involution which we have already seen. But the other one is a scalar product which is compatible with the multiplication.

#### Hall Inner Product

Recall an inner product is a function

$$\langle - | - \rangle : V \times V \rightarrow \mathbb{Q}$$

which is bilinear  $\langle u + v | w \rangle = \langle u | w \rangle + \langle v | w \rangle$  and the same on the other entry. For scalars the following behavior is expected  $\langle \lambda u | v \rangle = \langle u | \lambda v \rangle = \lambda \langle u | v \rangle$ . Recall that if the base field is the complex numbers, then the inner product is Hermitian.

**Definition 1.4.1.** We say that two vectors are orthogonal when  $\langle u | v \rangle = 0$ .

This gives us a possible decomposition of space into several components. Suppose that  $\{u_\lambda\}_{\lambda \in \text{Par}(n)}, \{v_\lambda\}_{\lambda \in \text{Par}(n)}$  are basis of  $\Lambda^n$ . So we would like a condition such as

$$\langle u_\lambda | v_\mu \rangle = \begin{cases} 0 & \lambda \neq \mu, \\ 1 & \lambda = \mu. \end{cases}$$

If we cap the dimension this says that  $\langle u | v \rangle$  is the usual dot product. But in infinite dimensions we don't have matrices. We'll call this basis dual to one another. If miraculously we have the same basis, then this basis is orthonormal.

**Definition 1.4.2** (Phillip Hall). The Hall inner product is defined so that  $\langle m_\lambda | h_\mu \rangle = \delta_{\lambda\mu}$ .

By defining the product on two basis, we have defined it for all other elements by bilinearity.

**Lemma 1.4.3.** *The Hall inner product is symmetric.*

**Theorem 1.4.4.** *The Hall inner product is positive definite, this is  $\langle f|f \rangle \geq 0$  and equality is achieved when  $f = 0$ .*

It's important to note that this statement is symmetric. However we are talking about an asymmetric definition. Last, before proving the statement we need a criteria for dual bases. But importantly, recall the result from last lecture: 1.3.7

**Theorem 1.4.5.** *If  $u_\lambda, \{v_\mu\}$  are dual, then  $\sum_\lambda u_\lambda v_\lambda = \Omega$ .*

**Proof**

Fix a partition of  $n$ , then

$$\delta_{\lambda\mu} = \langle m_\lambda | h_\mu \rangle = \left\langle \sum_{\rho \vdash n} \alpha_{\lambda\rho} u_\rho \middle| \sum_{\tau \vdash n} \beta_{\mu\tau} v_\tau \right\rangle = \sum_{\rho, \tau} \alpha_{\lambda\rho} \beta_{\mu\tau} \langle u_\rho | v_\tau \rangle.$$

We want  $\langle u_\rho | v_\tau \rangle = \delta_{\rho\tau}$ , to that effect name  $A_{\rho\tau}$  the matrix whose entries are  $\langle u_\rho | v_\tau \rangle$ .

As  $u$  and  $v$  are dual bases, we have that  $A = \text{id}$ . Thus  $I = \alpha\beta^\top$  and now  $\delta_{\rho\tau} = \sum \alpha_{\lambda\rho} \beta_{\lambda\tau}$ . We are now going to use the hypothesis and the interpretation of  $m, h$  in the  $u, v$  basis. We have

$$\Omega = \sum \left( \sum \alpha u \right) \left( \sum \beta v \right) = \sum \left( \sum \alpha \beta \right) uv = \sum uv$$

so the inner sum must be one and thus we are done.

**Corollary 1.4.6.** *For the Hall inner product it holds that  $\langle p_\lambda | p_\mu \rangle = z_\lambda \delta_{\lambda\mu}$ .*

The key is to recall that  $p_\lambda$  is an eigenfunction of  $\omega$ . Also 1.3.5. By using a power-sum decomposition it is possible to prove that the Hall inner product is positive definite.

**Corollary 1.4.7.** *The  $\omega$  involution is orthogonal with respect to  $\langle - | - \rangle$ . This is  $\langle \omega f | \omega g \rangle = \langle f | g \rangle$ .*

Once again, the idea is to transfer to power-sum and use the fact that it's an eigenfunction.

## 1.5 Interim 1

**Theorem 1.5.1** (Fundamental Theorem of Sym. Fnc. Thry.). *Every symmetric function can be written uniquely in the form  $\sum_\lambda c_\lambda e_\lambda$  with  $c_\lambda \in \mathbb{Q}$ .*

## 1. SYMMETRIC FUNCTIONS

---

There are at least two proofs if not more of this fact. The first comes from Maria Gillespie's blog which Mark Haiman presented to her.

### Proof

It suffices to prove the transition matrix between  $m$  and  $e$  is invertible.

For proof 2 read [10] pg. 290. Proof 3 in another Maria post

## 1.6 Day 5 | 20230130

**Exercise 1.6.1.** Compute  $\omega(s_{(3,1)})$ .

### Answer

We have that By Jacobi-Trudi

$$s_{(3,1)} = \det \begin{pmatrix} h_3 & h_4 \\ 1 & h_1 \end{pmatrix} = h_{(3,1)} - h_4.$$

Using the omega involution, we get

Recall that  $\omega : h_n \leftrightarrow e_n, \omega p_k = (-1)^{k-1} p_k$ . We have the following questions, where do  $m$  and  $s$  map to? Also

$$\langle m|h \rangle = \delta, \langle p|p/z \rangle = \delta,$$

but what are  $e$  and  $s$  dual to?

**Definition 1.6.2.** We call  $\omega m_\lambda = f_\lambda$  the forgotten basis.

There's not much we could say about them, they are not Schur positive and there's no patterns.

### Dual to $e$

Recall  $\omega$  is an isometry, so  $\langle \omega f | \omega g \rangle = \langle f | g \rangle$ , so

$$\langle e_\lambda | ? \rangle = \langle h_\lambda | \omega ? \rangle = \delta_{\lambda\mu}.$$

Since  $\langle h|m \rangle = \delta$ , then applying  $\omega$  again we get that  $\langle e_\lambda | f_\mu \rangle = \delta_{\lambda\mu}$ .



## RSK algorithm

We want to show two things:

$$\omega s_\lambda = s_{\lambda^\tau}, \langle s_\lambda | s_\mu \rangle = \delta_{\lambda\mu}.$$

**Proposition 1.6.3.** *It holds that*

$$\sum_{\lambda} s_{\lambda}(\underline{x}) s_{\lambda}(\underline{y}) = \Omega = \sum_{\lambda} m_{\lambda}(\underline{x}) h_{\lambda}(\underline{y})$$

### Proof

The sum on the left is

$$\sum_{(S,T) \text{ SSYT}} x^S y^T$$

so we will study pairs  $(S, T)$  of SSYT of the same shape to show that they're equal to the sum on the right.

algorithm: process of doing the bijection.

The RSK bijection takes a pair  $(S, T)$  of SSYT of the same shape and it maps it to “two-line arrays” of length  $n$ .

**Definition 1.6.4.** A two-line array is a matrix in  $\mathcal{M}_{2 \times n}(\mathbb{Z}_{\geq 0})$  such that

- i) The bottom row is weakly increasing.
- ii) If  $b_i = b_{i+1}$ , then  $a_i \leq a_{i+1}$ , where  $a$ 's are the top row and  $b$ 's the bottom row.

**Example 1.6.5.** Consider the matrix

$$\begin{pmatrix} 1 & 1 & 2 & 1 & 4 & 2 & 3 & 1 & 2 \\ 1 & 1 & 1 & 2 & 2 & 3 & 3 & 4 & 4 \end{pmatrix}$$

Within “blocks”, there is a weak increment. From right-to-left we will find a pair of SSYT. We will “insert” top row letters from left-to-right.

(a) Place 1st letter  $\boxed{1}$

(b) For each letter, if it can go at the end of last row, put it there

$$\boxed{1} \boxed{1} \leftarrow 2, \boxed{1} \boxed{1} \boxed{2} \leftarrow 1$$

but one can't go after 2.

## 1. SYMMETRIC FUNCTIONS

---

- (c) Otherwise if inserting  $b_1$ , let  $c$  be the leftmost  $> b$ , “bump  $c$ ”, then insert  $c$  into the next row.

1	1	1
2		

For the bottom row, place in a new square at each step to form a “recording tableau”. The recording tableau always matches the shape of the insertion one. The first three

steps lead to 

1	1	1
2		

 in the recording one. But in the fourth step we get 

1	1	1
2		

. The next step leads us to

1	1	1	4
2			

, 

1	1	1	2
2			

then in insertion, 2 bumps 4 and 4 doesn't bump 2 on next row, so we get

1	1	1	2
2	4		

, 

1	1	1	2
2	3		

The three is no problem so

1	1	1	2	3
2	4			

, 

1	1	1	2	3
2	3			

then the next one bumps out the 2, the 2 bumps the 4 on the second row to get

1	1	1	1	3
2	2			
4				

, 

1	1	1	2	3
2	3			
4				

Finally

1	1	1	1	2
2	2	3		
4				

, 

1	1	1	2	3
2	3	4		
4				

.

Why do we get SSYT. The insertion tableau gives us the question, can we make a column non-increasing? No, we are always bumping something bigger. Imagine we bump  $c > b$  with  $b$ , then  $c$  replaces something that goes to the left.

$\leq$	$b$	$c$

 $\Rightarrow$ 

	$b$	
	$d$	

and  $d > c$  so it bumps something else. The recording tableau is also a SSYT. Let us prove it.

**Lemma 1.6.6** (Key Lemma 1). *The insertion path (sequence of squares that are bumped) moves up and weakly left.*

**Lemma 1.6.7** (Key Lemma 2). *If  $a \leq b$  and  $T$  is a SSYT, computing*

$$T \leftarrow \boxed{a} \leftarrow \boxed{b},$$

*the insertion path of  $a$  in  $T$  lies strictly left of the insertion path of  $b$  in  $T \leftarrow \boxed{a}$ .*

**Proof**

We will do induction on the rows with an example.

**Example 1.6.8.** Consider

1	1	1	2	2	3
2	2	3	3	4	
3	3	5	5		
4	4				

Inserting 1 we bump the 2, then the 3 and finally the 5. We get

1	1	1	<i>o</i>	2	3
2	2	<i>t</i>	3	4	
3	3	<i>t</i>	5	5	
4	4	<i>f</i>			

so inserting the 2 we bump 3,4,5. And they will be to the side of the last sequence.

## 1.7 Day 6 | 20230201

**Exercise 1.7.1.** Apply RSK to  $\begin{pmatrix} 3 & 2 & 4 & 1 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}$

**Answer**

We get 

1	4	5
2		
3		

, 

1	3	5
2		
4		

.

Notice that we got STANDARD Young tableau. So to prove it's a bijection we will begin with all different numbers.

## 1. SYMMETRIC FUNCTIONS

---

**Lemma 1.7.2.** *The RSK bijection is a bijection between pairs of standard Young tableaux of the same shape and “permutations” ( $2 \times n$  matrices whose rows are permutations.)*

To prove it’s a bijection we will find an inverse by reversing the process. Look at the recording tableau, we will bump out the largest number. We will take  $S$  as the recording tableau. Then we start with the spot on  $S, T$  which corresponds to largest label in  $S$ .

- ◊ If  $b$  is the item in such a square we “un-bump” it.
  - If in bottom row, just remove.
  - Else, let  $c$  be the rightmost entry in row below  $b$  that is less than  $b$ . Then replace  $b$  with  $c$  and repeat the process with  $c$  until the letter that is removed is done by the just removing it.

Then we add the two letters to the matrix from right-t-left.

With the original tableau we remove the 5 and the 5 to get

$$\begin{array}{|c|c|} \hline 1 & 4 \\ \hline 2 & \\ \hline 3 & \\ \hline \end{array}, \begin{array}{|c|c|} \hline 1 & 3 \\ \hline 2 & \\ \hline 4 & \\ \hline \end{array}$$

then the 4 indicates that in  $T$  we must “un-bump” the 3. The three un-bumps the 2, the 2 to the 1 so that we get

$$\begin{array}{|c|c|} \hline 2 & 4 \\ \hline 3 & \\ \hline \end{array}, \begin{array}{|c|c|} \hline 1 & 3 \\ \hline 2 & \\ \hline \end{array}.$$

Now we get the matrix  $\begin{pmatrix} x & x & x & 1 & 5 \\ x & x & x & 4 & 5 \end{pmatrix}$  and removing the 3 from  $S$  just removes the 4 from  $T$  as it is in the bottom row.

Now as this two sets are in bijection, this means that they have the same size.

**Corollary 1.7.3.** *Let  $f^\lambda$  be the number of standard Young tableau of shape  $\lambda$ . Then*

$$\sum_{\lambda \vdash n} (f^\lambda)^2 = n!.$$

We will generalize one step at a time. Let us now assume that  $T$  is semi-standard. On the matrix, we will have that the top row is now random, but the bottom row is still from 1 to  $n$ .

**Lemma 1.7.4** (Schensted). *There is a bijection between  $(S, T)$ ,  $S$  is standard,  $T$  is SSYT, and words of length  $n$ .*

**Example 1.7.5.** Consider the matrix  $\begin{pmatrix} 2 & 1 & 3 & 1 & 3 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}$  which returns the two Young tableau

$$\begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 3 & \\ \hline \end{array}, \quad \begin{array}{|c|c|c|} \hline 1 & 3 & 5 \\ \hline 2 & 4 & \\ \hline \end{array}.$$

The proof of the inverse is similar but when un-bumping, we must bump the rightmost entry *strictly* smaller than  $b$ . But we don't need this, we will do it more creatively.

**Definition 1.7.6.** Suppose  $T$  is a Young tableau. Then

- i) The reading word of  $T$   $rw(T)$  is the concatenation of rows from top to bottom.
- ii) The standarization of an SSYT  $T$ ,  $std(T)$ , is the unique SYT with same relative order of entries, ties broken with "reading order".
- iii) The standarization of a word is similar

In the previous example, the reading word is

$$\begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 3 & \\ \hline \end{array} \rightarrow 23113.$$

The standarization are as follows:

$$\begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 3 & \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 1 & 2 & 5 \\ \hline 3 & 4 & \\ \hline \end{array}, \quad 23113 \rightarrow 34125.$$

We can standarize the matrix

$$\begin{pmatrix} 2 & 1 & 3 & 1 & 3 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix} \rightarrow \begin{pmatrix} 3 & 4 & 1 & 2 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}$$

and this matrix corresponds to the pair  $(S, T)$  where  $T$  is  $\begin{array}{|c|c|c|} \hline 1 & 2 & 5 \\ \hline 3 & 4 & \\ \hline \end{array}$ . In essence, the following diagram commutes

$$\begin{array}{ccc} (S, T) & \xleftarrow{RSK} & 21313 \\ \text{\scriptsize std} \downarrow & & \downarrow \\ (S, T') & \xleftarrow{RSK} & 31425 \end{array}$$

## 1. SYMMETRIC FUNCTIONS

**Definition 1.7.7.** Given a content  $\mu = (\mu_1, \dots, \mu_k)$  with  $\sum \mu_k = n$  (not nec. partition). Then the de-standarization with respect to  $\mu$  of a SYT  $T$  is a SSYT  $T'$  such that  $\text{std}(T') = T$ .

In this case

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 5 \\ \hline 3 & 4 & \\ \hline \end{array} \xrightarrow{\text{std}^{-1}(2,1,2)} \begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 3 & \\ \hline \end{array}.$$

Recall now lemma 1.6.7 about consecutive insertions.

### The Full RSK

We are now going to prove that there is an inverse to the original RSK function. Consider the following example

$$\begin{pmatrix} 1 & 1 & 2 & 1 & 4 & 2 & 3 & 1 & 2 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{pmatrix} \rightarrow \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 2 \\ \hline 2 & 2 & 3 & & \\ \hline 4 & & & & \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|c|} \hline 1 & 2 & 3 & 5 & 7 \\ \hline 4 & 6 & 9 & & \\ \hline 8 & & & & \\ \hline \end{array}$$

The matrix  $\begin{pmatrix} 1 & 1 & 2 & 1 & 4 & 2 & 3 & 1 & 2 \\ 1 & 1 & 1 & 2 & 2 & 3 & 3 & 4 & 4 \end{pmatrix}$  can be standarized to our word matrix. Then

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 2 & 3 \\ \hline 2 & 3 & 4 & & \\ \hline 4 & & & & \\ \hline \end{array}$$

the table  $\begin{array}{|c|} \hline 4 \\ \hline \end{array}$  also standarizes to the word table.

## 1.8 Day 7 | 20230203

**Exercise 1.8.1.** Expand  $h_{(3,2)}$  in Schur basis.

**Answer**

This is  $s_{(3,2)} + s_{(4,1)} + s_{(5)}$ .

Recall that  $(s_\lambda)$  form an orthonormal basis and  $m$  and  $h$  are dual basis. This means that if  $f$  is a symmetric function then

$$f = \sum_{\lambda} c_{\lambda} s_{\lambda} \Rightarrow c_{\lambda} = \langle f | s_{\lambda} \rangle, \quad f = \sum_{\lambda} a_{\lambda} m_{\lambda} \Rightarrow a_{\lambda} = \langle f | h_{\lambda} \rangle.$$

Lets suppose now that  $f$  is any homogenous symmetric function. We will calculate the coefficient of  $s_{\lambda}$  in an  $h_{\mu}$  expansion:

$$\langle h_{\mu} | s_{\lambda} \rangle = \langle s_{\lambda} | h_{\mu} \rangle$$

and we can interpret this as the coefficient of  $m_\mu$  in  $s_\lambda$ . This amount is precisely the Kostka coefficient  $K_{\lambda\mu}$ . Thus we have the formula  $h_\mu = \sum_\lambda K_{\lambda\mu} s_\lambda$ .

### Properties of the Schur functions

We wish to show that  $\langle s_\lambda | s_\mu \rangle = \delta_{\lambda\mu}$  and  $\omega s_\lambda = s_{\lambda^\tau}$ .

**Proposition 1.8.2.**  $\sum_\lambda s_\lambda(\underline{x}) s_\lambda(\underline{y}) = \sum_\lambda m_\lambda(\underline{x}) h_\lambda(\underline{y})$

#### Proof

Expanding the sum on the left we obtain

$$\sum_\lambda s_\lambda(\underline{x}) s_\lambda(\underline{y}) = \sum_\lambda \lambda \left( \sum_{T \in SSYT(\lambda)} x^T \right) \left( \sum_{S \in SSYT(\lambda)} y^S \right) = \sum_{(T,S), SSYT \text{ same shape}} x^T y^S$$

This is basically an RSK pair and this correspond to two-line arrays, so this sum could be the same as summing over them. Thus this is

$$\sum_{\text{2line arrays}} x_{a_1} \dots x_{a_n} y_{b_1} \dots y_{b_n}.$$

We will now find the coefficient of  $m_\lambda(\underline{y})$  in this expansion and show that it is  $h_\lambda(\underline{x})$

What are all the ways to obtain  $y_1^{\lambda_1} \dots y_k^{\lambda_k}$ ?

$$\begin{pmatrix} a_1^{(1)} & \dots & a_k^{(1)} & a_1^{(2)} & \dots & a_k^{(2)} & \dots \\ 1 & 1 & 1 & 2 & 2 & 2 & \dots \end{pmatrix}$$

And note that  $a_1^{(i)} \leq \dots \leq a_{\lambda_i}^{(i)}$  for all  $i$ , so the coefficient is

$$\sum_{(a^{(i)}) \text{ valid tuples}} x_{a_1^{(1)}} \dots x_{a_{\lambda_k}^{(k)}}$$

but this factors as

$$\prod_{i=1}^k \sum_{a_1^{(i)} \leq \dots \leq a_{\lambda_i}^{(i)}} x_{a_1^{(i)}} \dots x_{a_{\lambda_i}^{(i)}}.$$

We can split this because the choices are independent of the blocks and then multiply the functions together. The last term is  $h_{\lambda_i}$  and the product is  $h_\lambda$ .

## 1. SYMMETRIC FUNCTIONS

If  $(T, S)$  RSKs inverse to  $\begin{pmatrix} 1 & 3 & 2 \\ 1 & 1 & 2 \end{pmatrix}$  then  $x^T y^S$  is  $x_1 x_3 x_2 y_1 y_1 y_2$ .

**Corollary 1.8.3.**  $\omega s_\lambda = s_{\lambda^\tau}$ .

### Proof

It suffices to show  $\langle s_{\lambda^\tau} | e_\mu \rangle = K_{\lambda\mu}$  because  $\langle s_\lambda | h_\mu \rangle = K_{\lambda\mu}$  which implies that  $\langle \omega s_\lambda | e_\mu \rangle = K_{\lambda\mu}$ .

In other words, we wish to show that the coefficient of  $s_\lambda$  in  $e_\mu$  is  $K_{\lambda^\tau\mu}$ , the number of *SSYT* shape  $\lambda^\tau$ , content  $\mu$ .

**CONT**

### Pieri Rule

**Definition 1.8.4.** A skew shape is a diagram formed by subtracting a smaller Young diagram from a larger one.

A horizontal strip is a skew shape where no two boxes are in the same column. Similar a vertical strip doesn't have boxes in the same row.

**Example 1.8.5.** Suppose  $\lambda = (5, 4, 4, 1)$  and  $\mu = (4, 2, 2)$ . Then

$$\lambda = \begin{array}{cccccc} & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{array}, \quad \mu = \begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array}$$

so  $\lambda/\mu$  is INSERT DIAG. Not horizontal nor vertical.

In a Young tableau, the biggest number forms a horizontal strip, so in general Young tableaux are made up of horizontal strips.

$$\begin{array}{cccccc} 4 & & & & & \\ 3 & 4 & 4 & & & \\ 2 & 2 & 3 & 3 & & \\ 1 & 1 & 1 & 2 & 3 & 4 \end{array}$$

**Theorem 1.8.6 (Pieri).** Let  $r \in \mathbb{N}$ , then

$$e_r s_\lambda = \sum_{\rho/\lambda \text{ vert. strip size } r} s_\rho$$

$$h_r s_\lambda = \sum_{\rho/\lambda \text{ horiz. strip size } r} s_\rho$$



This is basically all the ways to fill up the shapes.

Proof

$$h_{rs\iota a} = s_{(r)} s_{\lambda} = \left( \sum_{T \in SSYT((r))} x^T \right) \left( \sum_{S \in SSYT(\lambda)} x^S \right)$$

**Example 1.8.7.**  $h_{3s(3,1)}$  is  $x^T x^S$  is inserting the boxes of  $T$  one at a time in  $S$ .

$$\begin{array}{|c|} \hline 3 \\ \hline \end{array} \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline \end{array} \leftarrow \begin{array}{|c|c|c|} \hline 1 & 1 & 2 \\ \hline \end{array} = \begin{array}{|c|c|c|c|} \hline 3 & 2 & 3 & \\ \hline \end{array} \begin{array}{|c|c|c|c|} \hline 1 & 1 & 1 & 2 \\ \hline \end{array}$$

so by 1.6.6 about insertion path, the new squares are a horizontal strip which is the  $s_{\rho}$  in the Pieri rule. Unbumping we recover **something**.

## 1.9 Day 8 | 20230206

**Exercise 1.9.1.** Apply RSK to 82357146 and 62235124.

Answer

$$\begin{array}{|c|} \hline 8 \\ \hline \end{array} \begin{array}{|c|c|c|c|} \hline 2 & 3 & 5 & 7 \\ \hline \end{array}, \begin{array}{|c|} \hline 2 \\ \hline \end{array} \begin{array}{|c|c|c|c|} \hline 1 & 3 & 4 & 5 \\ \hline \end{array}$$

then 1 bumps the 2, 2 bumps 8

$$\begin{array}{|c|} \hline 8 \\ \hline \end{array} \begin{array}{|c|c|c|} \hline 2 & 5 & 7 \\ \hline \end{array} \begin{array}{|c|} \hline 6 \\ \hline \end{array} \begin{array}{|c|c|c|c|} \hline 2 & 7 & 8 & \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|} \hline 1 & 3 & 4 & 6 \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 1 & 3 & 4 & 5 \\ \hline \end{array}$$

The next one standarizes to the last string. The same recording table but we get for insertion

$$\begin{array}{|c|} \hline 6 \\ \hline \end{array} \begin{array}{|c|c|c|} \hline 2 & 3 & 5 \\ \hline \end{array} \\ \begin{array}{|c|c|c|c|} \hline 1 & 2 & 2 & 4 \\ \hline \end{array}$$

## Consequences of RSK

We will talk about increasing and decreasing subsequences.

## 1. SYMMETRIC FUNCTIONS

**Definition 1.9.2.** A longest increasing subsequence of a word  $w \in \mathbb{N}^n$  is a subsequence  $w_{i_1} \leq \dots \leq w_{i_\ell}$  with  $i_1 < \dots < i_\ell$  such that  $\ell$  is as large as possible. We will write  $\ell(w)$  to be the length of the longest increasing subsequence.

A longest decreasing subsequence of a word is  $w_{i_1} > \dots > w_{i_d}$  with  $i_1 < \dots < i_d$ . In this case  $d(w)$  is the longest decreasing.

**Example 1.9.3.** In the case of 82357146, we have 2357, 2356, 146, 2346. Notice that this is the length of ?? of the Young tableau. For decreasing we have 821, 831, ..., the height of the Young tableau is the longest decreasing subsequence.

**Theorem 1.9.4.** Suppose  $w$  is a word,  $S = \text{ins}(w)$  is the insertion tableau through RSK and  $\lambda = \text{sh}(S)$  is the shape of the table. Then  $\ell(w) = \lambda_1$  and  $d(w) = \lambda_1^T$ .

To prove this we will develop some tools.

**Lemma 1.9.5.** For a tableau  $T$ ,  $\text{ins}(\text{rw}(T)) = T$ .

8			
2	5	7	
1	3	4	6

The reading word of 

1	3	4	6
---	---	---	---

 is 82357146 which inserts to the same table precisely.

*Remark 1.9.6.* The column reading word also works! For this table it's 82153746. We get a bunch of decreasing subsequences. 821 creates the first column by bumping, then 53 creates the second column and so on.

Let's analyze the longest increasing subsequence of the reading word. Clearly we can get the bottom row as a longest subsequence, but looking in the reading order we need to go to the right. Going down decreases!

**Lemma 1.9.7.** If  $\lambda = \text{sh}(T)$  then  $\ell(\text{rw}(T)) = \lambda_1$  and  $d(\text{rw}(T)) = \lambda_1^T$ .

### Proof

Given an entry  $a \in T$ , let  $b \in T$  such that  $a <_{\text{ro}} b$ . Then  $b$  is in a column to the right of  $a$ , this means that

$$\ell(\text{rw}(T)) \leq \#\text{columns} = \lambda_1.$$

The bottom row is an example of a subsequence where the length is achieved. So equality holds.

For decreasing it's equivalent. Now, how do we tell when two words have the same insertion tableau?

**Example 1.9.8.** In the case of all permutations in  $S_3$  we have that some are equivalent  
**FILL**

### Knuth equivalence

**Definition 1.9.9.** A Knuth move on a permutation swaps two letters  $a, c$  if  $a < b < c$  (reading order) and one of consecutive subsequences  $acb, cab, bac, bca$  appears in the word.

Two words are Knuth-equivalent if they differ by a sequence of Knuth-moves.

In the first case,  $b$  is between  $a, c$  and those are always together.

**Proposition 1.9.10.** *Knuth equivalence defines an equivalence relation on  $S_n$ .*

**Theorem 1.9.11.** *Two words  $\pi, w$  are Knuth-equivalent iff  $\text{ins}(w) = \text{ins}(\pi)$ .*

**Example 1.9.12.** In size 4, 1234 is in its own class because we don't have any Knuth moves available. Same thing happens with 4321.

Consider 1243, if we apply Knuth moves we can get

◇ 1423

◇ 4123

All of these have the insertion tableau 

4
1 2 3

 whose reading word is 4123.

For the tableau 

2 4
1 3

, its reading word is 2413. Applying Knuth moves we get only 2143, which is the column reading word.

The tableau 

3 4
2 1

's equivalence class also has size 2.

**Proposition 1.9.13.** *If two tableau have the same shape, their equivalence classes have the same size.*

We are seeking to prove  $\ell(w)$  is invariant under Knuth moves. This will imply the theorem 1.9.4 because once we know that things have the same insertion tableau and the reading word has the same longest increasing subsequence length.

**1.10 Day 9 | 20230208****Exercise 1.10.1.** Insert  $f, g$  and then  $c$  into

$k$				
$e$	$i$	$j$		
$a$	$b$	$d$	$h$	$l$

and then  $f, c$  and then  $g$ .**Example 1.10.2.** The Knuth equivalence class of words whose insertion tableau is

3	4	
1	2	5

The reading word is 34125 and we can Knuth-move it. The 341 can switch into 314 (this has the form  $bac$ ). From that one we can switch 2 and 5 to get 31452. Once again with 314 we get 34152 and 34512.

In total we have 5 elements.

**Proposition 1.10.3.** *The size of the Knuth equivalence class whose insertion tableau is  $T$  with shape  $\lambda$  is  $\#SYT(\lambda)$ .***Proof**

We have one permutation in the Knuth equivalence class for every recording tableau  $S$  that can be paired with  $T$ .

3	4	
1	2	5

The Knuth equivalence class of  $\begin{smallmatrix} 3 & 4 \\ 1 & 2 & 5 \end{smallmatrix}$  can be identified by RSK with the pairs  $(T, S)$  and  $S$  varies through all SYT of corresponding shape.

Also, recall that by that hook-length formula we have that

$$\#SYT(\lambda) = \frac{|\lambda|!}{\prod_{\text{hooks} \subseteq T} \text{size hooks}}.$$

**Theorem 1.10.4.** *Two permutations  $\pi, w$  have the same insertion tableau if and only if  $\pi$  is Knuth-equivalent to  $w$ .*

**Proof**

By induction on the length, we can assume  $\pi, w$  differ by a single Knuth-move on the last 3 letters. We separate into cases:

i) Want

$$T' \leftarrow b \leftarrow c \leftarrow a = T' \leftarrow b \leftarrow a \leftarrow c$$

Note that  $\text{IP}(b) < \text{IP}(c)$  by lemma 1.6.7 of consecutive insertions and  $\text{IP}(a)$  is *weakly left* of  $\text{IP}(b)$  from which holds  $\text{IP}(a)$  is strictly left of  $c$ 's. So we can switch order.

ii) In the other case we want

$$T' \leftarrow c \leftarrow a \leftarrow b = T' \leftarrow a \leftarrow c \leftarrow b.$$

$\text{IP}(a)$  is *weakly left* of  $c$ 's. If it's *strictly*, then we can switch, but otherwise the insertion paths of  $a$  and  $c$  collide. **CHECK NOTES**

Now on the other direction, we wish to show that two permutations with the same insertion tableau are Knuth-equivalent.

It suffices to show that they are Knuth-equivalent to the reading word. By induction of the size of the word, suppose  $\text{ins}(w') = T'$ . Then  $w' \sim \text{rw}(T')$  for  $w'$  of length  $n - 1$ .

Let  $w \in S_n$  with  $b = w_n$ . If  $T' = \text{ins}(w_1, \dots, w_{n-1})$ , by induction  $w_1 \dots w_{n-1} \sim \text{rw}(T') = (\text{first row}) \dots (\text{last row})$ .

**Example 1.10.5.** For the second case consider the table

3	8	9
1	4	7

and we insert 6, 2 then 5 but then 2, 6 and then 5. In the first case, **DUNNO**

In the second case consider

$$T' = \begin{array}{|c|c|c|c|} \hline 6 & & & \\ \hline 4 & 7 & & \\ \hline 1 & 2 & 5 & 8 \\ \hline \end{array}$$

## 1.11 Day 10 | 20230210

**Lemma 1.11.1.** The length of the longest increasing subsequence,  $\ell(w)$  is invariant under Knuth moves.

## Proof

Given an increasing subsequence, if a Knuth move changes two of its entries  $a < c$ , we have two cases:

i) Either  $b$  is to the right of  $ac$  so we get

$$\dots acb \dots d \dots \rightarrow \dots cab \dots d \dots$$

then replacing  $c$  with  $b$  gives an increasing subsequence of same length in a new word  $a < b < c$  by assumption so  $\underline{a} < \underline{b} < c < \underline{d}$  where  $d$  is the next element of the subsequence after  $c$ .

ii) We have  $b$  to the left so we have

$$\dots bac \dots d \dots \rightarrow \dots bca \dots d \dots$$

and the same proof shows that replacing  $a$  and  $b$  gives a new subsequence and a new subword of the same length.

Knuth equivalence is natural when it comes to increasing and decreasing subsequences. With the lemmas we have, we can now prove that  $d$  is the height of the insertion tableau.

*Remark 1.11.2.* Dual equivalence is finding 3 values  $acb$  so we can switch  $bca$ . For example

$$615342 \rightarrow 625341 \rightarrow 624351$$

and dual equivalent words have the same *recording tableau*.

The result we've been aiming for is

$$\ell(w) = \text{width of ins}(w) = \lambda_1$$

## Proof

Suppose  $T = \text{ins}(w)$  with  $\lambda = \text{sh}(T)$ . Then  $w \sim \text{rw}(T)$ , by the previous lemma we have

$$\ell(w) = \ell(\text{rw}(T)) = \lambda_1$$

and the last equality comes from Monday class. [Add references](#)

For decreasing subsequences we have the same argument. To wrap it up we have a theorem we have a theorem from Stanley [10]:

**Theorem 1.11.3.** A longest  $i$ -chain of increasing subsequences of  $w$  consists of:

- ◊ An increasing subsequence  $s_1$  of  $w$ .
- ◊ An increasing subsequence  $s_2$  of  $w \setminus s_1$ .
- ◊ (...) An increasing subsequence  $s_i$  of  $w \setminus s_1 \dots s_{i-1}$ .

Then the length of the longest  $i$ -chain is  $\lambda_1 + \lambda_2 + \dots + \lambda_i$ .

### Jeu de taquin

The phrase *jeu de taquin* means “teasing game”. This process is equivalent to RSK and insertion. As motivation, inserting  $\emptyset \leftarrow w$  and then insert  $\rho$  and finally  $\pi$ , then this is the same as

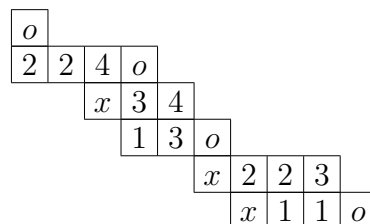
$$w \leftarrow \text{rw}(\text{ins}(\rho) \leftarrow \pi).$$

This means that there’s some *associativity* in this operation.

**Definition 1.11.4.** A skew-SSYT is a filling of the boxes of a skew shape (skew Ferrers diagram) with  $n \in \mathbb{N}$  such that rows are weakly increasing and columns are strictly increasing.

This is analogous to SSYT, so let’s see an example

**Example 1.11.5.**  $(9, 9, 5, 5, 3)/(7, 6, 3, 3)$ , but the initial partition could’ve been  $(10, 10, 6, 6, 4)$  or  $(100, 9, 9, 5, 5, 3)$ .



Notice that all rows and columns are adjacent, and there are no leftover squares.

**Definition 1.11.6.** A corner of a Young diagram  $\mu$  is a square of  $\mu$  at the top of its column, right of its row.

An inner corner of  $\lambda/\mu$  is a corner of  $\mu$ . While an outer corner of  $\lambda/\mu$  is a square outside  $\lambda$  that is just above its column (poss. o) just right of a row (poss. o).

*Remark 1.11.7.* Notice that adding an outer corner makes a valid shape for a partition.

The JdT game is defined through the following moves

## 1. SYMMETRIC FUNCTIONS

---

**Definition 1.11.8.** An inner slide into an inner corner  $x$  of a SSYT  $T$  of skew-shape  $\lambda/\mu$  is given by the following process:

- i) Compare squares  $a, b$  in the following shape

$a$	
$x$	$b$

If  $a \leq b$  or there's no  $b$ , slide  $a$  down. Else slide  $b$  left.

- ii) Repeat for the new location of  $x$  until it becomes an outer corner.

**Example 1.11.9.** In the previous example we can consider

$x$	2	2	3	
$x$	1	1	$o$	

and slide the 1 because if we slide 2 it's no longer SSYT. But then we get an empty square, so we keep going until it's a valid shape. The end result is

$x$	2	2	$o$	
	1	1	3	$o$

For a bigger example consider

2	4	4	5			
	3	3	3	6		
	1	2	2	3		
	$x$	1	1	2		

→

2	4	4	$o$			
	3	3	5	6		
	2	2	3	3		
	1	1	1	2		

*Remark 1.11.10.* A slide sends an SSYT to an SSYT. A case like

	3	
4	$x$	5

because there should be an entry there  $\leq 3$  and  $> 4$ .

We can continue applying the process to the following inner corners to get a SSYT.

**Definition 1.11.11.** The rectification of a skew SSYT is the result of performing JDT slides until we don't have inner corners left. We say that the shape is straight.

*Remark 1.11.12.* The rw of the skew-shape is the same as the rectification.



## 1.12 Day 11 | 20230213

Recall we have talked about the Jeu de Taquin. We observed that JDT slides send skew SSYT to other skew SSYT. To show compatibility with Knuth-equivalence, we want will require a more general definition of reading word because for example in the middle of a JDT we may have

3	4	7	8
	2	4	
	1	1	3 3

**Definition 1.12.1.** A reading word of any labeled set of boxes in the 1<sup>st</sup> quadrant is formed by reading the rows from top to bottom L to R with each row.

**Lemma 1.12.2.** If a skew SSYT  $S$  is obtained from another skew SSYT  $T$  by a sequence of JDT slides, then  $rw(S) \sim_K rw(T)$ .

### Proof

Assume  $S, T$  differ by one inner-slide. There are a couple of cases, we'll show that at each step of the slide the Knuth equivalence class of the reading word is unchanged. (Even when we haven't finished the slide.)

The first case is when we do a horizontal slide

$a$		$a$
$x$	$b$	$b$ $x$

and nothing happens here because the reading word is still  $ab$ . The interesting case is the vertical slide.

PROOF BY EXAMPLE

**Example 1.12.3.** Consider the tableau

$s$	$t$	$u$	$f$	$f$					
1	2	5	6	7	$T$				
		3	4	$x$	8	9			

the reading word changes from 12567 $T$ 3489 to 1256 $T$ 34789. We will apply Knuth moves to the reading word of the original tableau.

We need an algorithm to guarantee that we get from one point to another. It suffices to consider the subword 567 $T$ 348 on the smaller window

5	6	7	$T$
3	4	$x$	8

## 1. SYMMETRIC FUNCTIONS

---

Thinking in steps we want to move the 3 backwards past the 7, then 4 and finally pull the 10 back. We have

$$567T348 \rightarrow 5673T48$$

Notice that  $6 > 3$  because  $6 > 4 > 3$  and it's also less than 7, there's still something still past the 7 which we can use to move the 3. We get

$$5637T48 \rightarrow 56374T8$$

We use the 5 to pull the 3 out of the way, once again the 5 is there due to semi-standardness. We get

$$53674T8 \rightarrow 53647T8$$

and to move the  $T = 10$  we use the 8 which is below 10 due to semi-standardness. Applying a Knuth move with  $7T8$  we obtain

$$5364T78 \rightarrow 5634T78 \rightarrow 563T478 \rightarrow 56T3478$$

*Remark 1.12.4.* If we had more *stuff* we would have to take a longer subword.

We said the rectification was the end goal of a JDT, with this in hand we can *well*-define it.

**Definition 1.12.5.** The rectification of a SSYT is formed by performing inner JDT slides until we have a straight shape.

This is well defined because no matter the order of slides, the reading word of the rectification is equivalent to the reading word of the original tableau.

As the Knuth equivalence determines the insertion tableau ( $\text{rect}(T) = \text{ins}(\text{rw}(T))$ ) (knuth class = ins tableau thm.) So putting it all together we see it works.

**Example 1.12.6.** Rectify the tableau

$$\begin{array}{|c|c|} \hline 2 & 4 \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 5 \\ \hline \end{array} \begin{array}{|c|} \hline 3 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 2 & & \\ \hline 1 & 4 & 5 \\ \hline & & 3 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline 2 & \\ \hline 1 & 4 \\ \hline & 3 & 5 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array}.$$

However we can also do

$$\begin{array}{|c|c|} \hline 2 & 4 \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 5 \\ \hline \end{array} \begin{array}{|c|} \hline 3 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 2 & & \\ \hline & 4 & 5 \\ \hline 1 & 3 & \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 2 & 4 & 5 \\ \hline & 1 & 3 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|} \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array}.$$

The Knuth equivalence of the reading words of all tableau is the same! The other method is to take the reading word of the original tableau 24153 and then insert it to the corresponding tableau.

Now we can completely replace insertion with rectification because we can make any word with skew-tableau.

**Corollary 1.12.7.** *The rectification of the skew tableau where  $w$  is any word is  $\text{ins}(w)$ .*

**Example 1.12.8.** Consider the tableau

5			
4	6	7	
1	2	4	5
			3

we will rectify it and see it is the same as bump-inserting the 3 into the tableau. We get

5			
4	6	7	
1	2	4	5
			3

 $\rightarrow$ 

5			
4	6	7	
1	2	4	
		3	5

 $\rightarrow$ 

5	6		
4	4	7	
1	2	3	5

## DO THE SLIDING

Then doing the diagonal process is intuitively inserting by this process.

**Definition 1.12.9.** We define the product of two tableau  $T, U$  as the rectification of the skew tableau formed by connecting the lower right corner of  $T$  to the upper left of  $U$ .

Equivalently

$$T \circ U = \text{rect}(T \leftarrow \text{rw}(U)).$$

This gives us an associative operation because of the Knuth-equivalence. In consequence the set of tableaux is a monoid, as the identity element is the empty tableau.

We define it as the Plactic monoid, the set of SSYT's of straight shape with  $\circ$ .

It's interesting to look at the Plactic monoid in terms of words as well.

**Definition 1.12.10.** The Plactic monoid is the  $\{\text{words}\} / \sim$  with the concatenation of words as the operation and  $\sim$  is Knuth-equivalence.

**Example 1.12.11.** If  $w = [2131], v = [2213]$ , then

$$wv = [21312213] = [21132213].$$

For the next class we will talk about skew Schur functions. We will build up to writing skew functions in terms of the ordinary ones.

**1.13 Day 12 | 20230215****Skew Schur functions****Definition 1.13.1.**

$$s_{\lambda/\mu} = \sum_* x^T$$

where  $x^T = x^{\#1's} x^{\#2's} \dots$ . This is still a symmetric function by the same proof as for  $s_\lambda$ .

**Example 1.13.2.** Compute  $s_{(3,2)/(1)}$  in terms of  $m$  basis **PHOTO**

But instead we can see that this function is also  $s_{(2,2)} + s_{(3,1)}$ . It turns out that skew Schur functions are Schur positive. Recall that this means that in the Schur basis expansion, all of its coefficients are positive integers.

**Theorem 1.13.3.**  $s_{\lambda/\mu}$  is Schur-positive,

$$s_{\lambda/\mu} = \sum_{\nu \vdash |\lambda| - |\mu|} c_{\mu, \nu}^\lambda s_\nu$$

The coefficients are called the Littlewood-Richardson coefficients.

We can use the Littlewood-Richardson coefficients to count certain Young tableaux. However another way to compute coefficients which uses a similar rule is the *Knutson-Tao puzzles*.

**Littlewood-Richardson rule**

**Definition 1.13.4.** A word  $w$  of positive indices is (reverse) ballot, Yamanouchi, or lattice if every suffix  $w_i w_{i+1} \dots w_n$  has partition content. This is that reading the word from right-to-left the number of 1's is greater the 2's and so on.

$$\#1's \geq \#2's \geq \#3's \geq \dots$$

**Example 1.13.5.** Consider the word 341231211, this is Yamanouchi, while 21433231211 is not Yamanouchi.

**Definition 1.13.6.** A skew tableau is called Littlewood-Richardson if its reading word is a Yamanouchi word.

**Theorem 1.13.7.** The Littlewood-Richardson coefficient is the number of Littlewood-Richardson tableaux of shape  $\lambda/\mu$  and content  $\nu$ .

**Example 1.13.8.** Let us find some Littlewood-Richarson coefficients. Consider

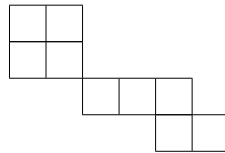
$$\lambda/\mu = \begin{array}{|c|c|} \hline & \\ \hline & \\ \hline \end{array}, \nu = (2, 2)$$

so we need to find L-R tableaux of shape  $\lambda/\mu$  with content  $(2, 2)$  (2 ones and 2 twos). The only possibilities are

$$\begin{array}{|c|c|} \hline 2 & 2 \\ \hline 1 & 1 \\ \hline \end{array}, \quad \text{and} \quad \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 1 & 2 \\ \hline \end{array}$$

and the latter's reading word is 1212 which is not Yamanouchi because it ends in a 2 and not a 1.

Let us now find the number of L-R tableaux with shape



and content  $(4, 3, 2)$ . We have the following tableaux

$$\begin{array}{|c|c|} \hline 3 & 3 \\ \hline 1 & 2 \\ \hline \end{array} \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline & 1 & 1 \\ \hline \end{array}, \quad \text{and} \quad \begin{array}{|c|c|} \hline 3 & 3 \\ \hline 2 & 2 \\ \hline \end{array} \begin{array}{|c|c|c|} \hline 1 & 1 & 2 \\ \hline & 1 & 1 \\ \hline \end{array}$$

This means that

$$c_{(4,2),(4,3,2)}^{(6,5,2,2)}.$$

Notice that by proving this result, we immediately get that the skew Schur functions are Schur positive. We will use crystal-base theory. This comes from the representation theory of  $\mathcal{U}_q(\mathfrak{sl}_n)$ .

A crystal of tableau sorts the monomials into a graph, let us for example consider

$$\begin{array}{|c|} \hline 2 \\ \hline 1 & 1 \\ \hline \end{array}$$

## 1.14 Day 13 | 20230217

### Crystals on words

Operating on words means we can operate on reading words and thus on tableau.

## 1. SYMMETRIC FUNCTIONS

---

**Definition 1.14.1.** For a word  $w$  of 1's and 2's, we define the raising operator  $E_1$  and lowering operator  $F_1$  by:

- ◊ Replacing all ones with right parenthesis and twos with left parenthesis.
- ◊ Then we pair off matching parenthesis.
- ◊  $E_1$  makes the first unmatched 2 to a 1 and  $F_1$  the last unmatched 1 to a 2.

**Example 1.14.2.** Consider the word

$$112212111122212 \rightarrow ))((()()))((()(($$

Applying  $E_1$  we get

$$112212111122212 \rightarrow 11221211111212 \rightarrow 112212111111211$$

and applying once again we get the empty word because there's no more match. We can see that applying  $F_1$  reverses this process. While applying  $F_1$  to our original word several times we get

$$222212112222212$$

before we get to the empty word.

**Definition 1.14.3.** For a word  $w \in \mathbb{Z}_{\geq 0}^n$ , then  $E_i$  is formed looking at the  $(i, i + 1)$ , treating them as  $(1, 2)$  and performing  $E_1$ .

This allows us to draw graphs on words where we connect after applying the operations.

The reason they are called raising and lowering is because in terms of content, the raise and lower the content:

$$222212112222212 \rightarrow (4, 11)$$

$$112212111122212 \rightarrow (8, 7)$$

$$112212111111211 \rightarrow (11, 4)$$

This also correspond to weight spaces in representations of  $\mathfrak{sl}_n$ .

*Remark 1.14.4.* Recall the Lie algebra of  $SL_2$  is

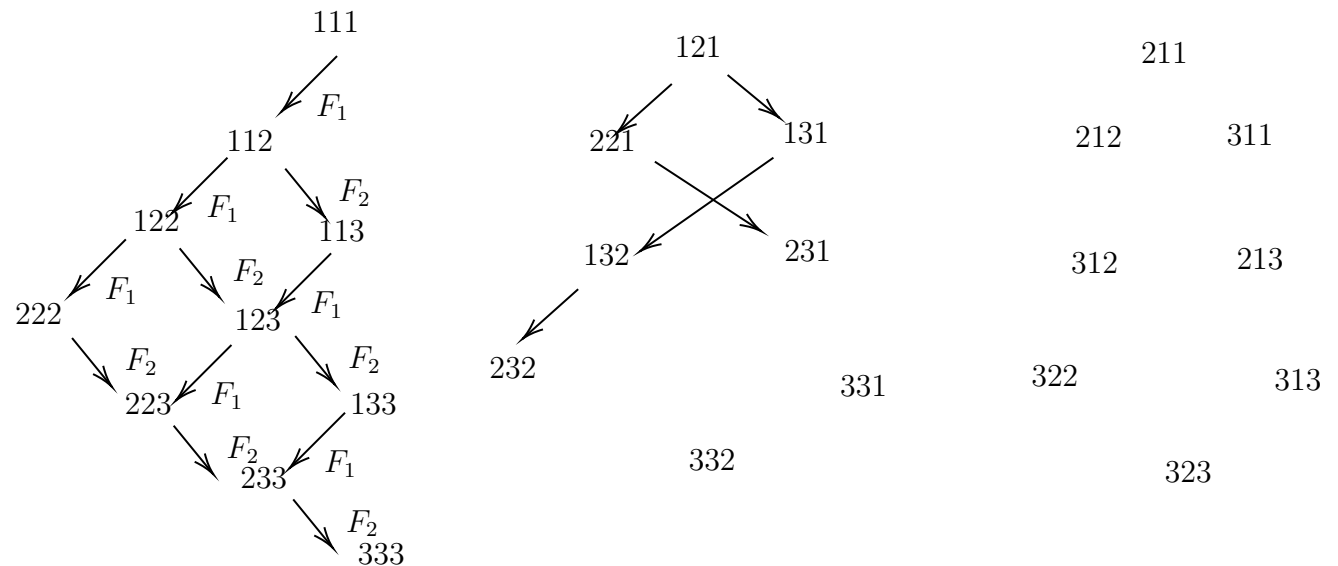
$$\mathfrak{sl}_2 = \{ A \in \mathcal{M}_2 : \text{tr}(A) = 0 \}.$$

This is an additive vector space with a Lie bracket, it's not closed under multiplication. The matrices with generate this space are

$$E = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad F = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad \text{and} \quad H = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

In general in  $\mathfrak{sl}_n$ ,  $E_i$  has a 1 on top of the  $i^{\text{th}}$  position on the diagonal.

**Example 1.14.5.** Consider the word 111 we get the following graph



Note that the words 121 and 211 both insert to

$$\text{ins}(121) = \text{ins}(211) = \begin{array}{|c|c|} \hline 2 & \\ \hline 1 & 1 \\ \hline \end{array}.$$

Also look at the fact that we have listed all the words of length 3. As an alternative definition, write all the words, match them by  $F$ 's then we get disjoint graphs. Each connected component gives us a Schur function.

*How many connected components do we have?*

If we have length  $n$ , the question is how many (reverse-)ballot words of length  $n$  are there? (Why ballot words?)

**Lemma 1.14.6** (Homework). *For a word  $w$*

$$E_i(w) = \emptyset \iff w \text{ is Yamanouchi.}$$

Assuming this, on length 4 we have the following Yamanouchi words:

- |          |          |          |
|----------|----------|----------|
| (a) 1111 | (c) 3211 | (e) 1121 |
| (b) 2111 | (d) 3121 | (f) 2211 |

## 1. SYMMETRIC FUNCTIONS

(g) 1321

(i) 1211

(h) 4321

(j) 2121

There are  $n \leq x \leq n!$  components. With the lemma comes the following definition:

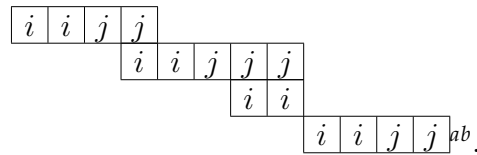
**Definition 1.14.7.** We say a word  $w$  has highest weight if  $E_i(w) = \emptyset$  for all  $i$ .

**Theorem 1.14.8.** By acting on the reading word we have that  $E_i, F_i$  are well defined on skew tableaux (and therefore on tableaux.)

We can always act on the reading word, but this action doesn't necessarily guarantee that the result is an SSYT.

### Proof

Look at the  $(i, i+1)$  in a tableau  $T$ . This will "look like"



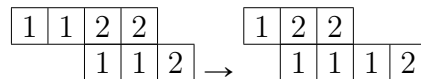
The columns "cancel" and some others do to, we can change the left most two  $i$ 's and the rightmost  $i(i+1)$ .

<sup>a</sup>We used this tableau when proving the Schur functions were symmetric.

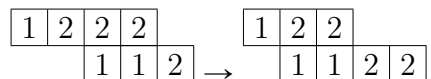
<sup>b</sup> $j$  is  $i+1$ , have to change ytableau package.

### Compatibility with Knuth moves and Jeu de Taquin

**Example 1.14.9.** Consider a skew tableau and an inner slide:



But applying  $F_1$  means that we get



### DRAW COMMDIAG

Compatibility means commutation at the level of diagrams.



### 1.15 Day 14 | 20230220

Today we will continue working on the Littlewood-Richardson rule. As a warm-up lets work on an example.

**Exercise 1.15.1.** Compute  $E_3$  of the tableau

5				
3	3	4	6	
2	2	3	4	
1	1	1	1	1

#### Answer

We can focus only on the sub skew tableau

3	3	4
	3	4

with reading word 33434, the leftmost un-paired 4 is the last one so we change the word to 33433 and then replace on the tableau to get

5				
3	3	4	6	
2	2	3	3	
1	1	1	1	1

Notice we do get a SSYT as we showed last time. We will now work on crystal operators.

**Theorem 1.15.2.**  $E_i, F_i$  commute with JDT slides.

It suffices to check this on 1 – 2 tableaux and  $E_1, F_1$ . This is because  $E_i, F_i$  only affect  $i, i + 1$ . (Look at example from last time) This example is the only case that we have to worry about because we can only have at most two rows. The only problematic case is a vertical slide like in the example.

But what about

1	2	2	2
	1	2	

 $\rightarrow$ 

1	2	2
1	2	2

it slides the same way after applying  $F_1$ :

2	2	2	2
	1	2	

 $\rightarrow$ 

2	2	2
1	2	2

## 1. SYMMETRIC FUNCTIONS

---

**Theorem 1.15.3.** *The operations  $E_i, F_i$  preserve the “being in the same Knuth-equivalence class”. In other words*

$$E_i(w) \sim E_i(v), \quad F_i(w) \sim F_i(v).$$

Since rectification is unique and that determines the Knuth-equivalence class, we can use the previous result.

### Answer

Consider the skew tableau  $D_w = \text{diag}(w)$  and  $D_v$ . Then

$$\text{ins}(w) = \text{ins}(v) \Rightarrow \text{rect}(D_w) = \text{rect}(D_v)$$

so

$$\begin{aligned} \text{rect}(E_i(D_w)) &= E_i(\text{rect}(D_w)) \\ &= E_i(\text{rect}(D_v)) \\ &= \text{rect}(E_i(D_v)) \end{aligned}$$

Therefore  $E_i(D_w) \sim E_i(D_v)$ .

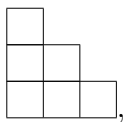
### Highest Weight

We will now work with highest-weight elements.

**Lemma 1.15.4.** *For a straight shape  $\lambda$ , the SSYT's with  $\text{sh}(\lambda)$  all lie in a single connected component of the crystal graph. (i.e. connecting via  $E_i, F_i$  we get a connected component.)*

*Moreover we get a unique tableau with highest weight ( $E_i(T) = \emptyset$ ). We claim that such a tableau has row  $i$  full of  $i$ 's.*

Take any shape



as the reading word is Yamanouchi, the first entry is a 1. Then all entries to the left are 1's by semi-standardness. The next rightmost element of the next row can't be a 3 by Yamanouchi so it must be a 2 and so on.

But why is everything in the same connected component?

If we have a tableau  $T$ , then applying  $E_i$  we *must* reach  $T_\lambda$ . Reversing this process for any  $T$  with  $F_i$ , we get all the tableau.

*Remark 1.15.5.* This method works with no restriction on the letters of the word.

**Corollary 1.15.6.** *Every connected component of a crystal of all tableaux of a given skew shape is isomorphic as a directed graph to a straight shape crystal obtained via JDT.*

With this key step we can compute a Schur function decomposition.

**Example 1.15.7.** Let us compute  $s_{(3,1)/(1)}$ . The crystal of this SSYT is obtained by writing the highest weight fillings of the skew shape:

$$\begin{array}{|c|} \hline 2 \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 1 \\ \hline \end{array}, \begin{array}{|c|} \hline 1 \\ \hline \end{array} \begin{array}{|c|c|} \hline 1 & 1 \\ \hline \end{array}.$$

We get the following crystals:

aaaaa

By rectifying with JDT we get INSERT DIAGRAMS

What this means is that adding monomials corresponding to crystal one we get  $s_{(2,1)}$  and the other crystal  $s_{(3)}$ .

As a shortcut, we draw the Yamanouchi ones, then rectify and that's it.

**Corollary 1.15.8.** *The coefficient of  $s_\nu$  is  $s_{\lambda/\mu}$  is the number of highest weight SSYT with  $\text{sh}(T) = \lambda/\mu$  that rectify to  $\text{sh}(\nu)$ .*

$$\# \text{hw sh}(\lambda/\mu) \quad \text{content} \quad \nu$$

this is exactly  $c_{\mu\nu}^\lambda$ .

This completes a proof of the Littlewood-Richardson rule.

## Products of Schur Functions

There's another Littlewood-Richardson Rule:

**Theorem 1.15.9.** *Suppose  $\mu, \nu$  are straight shapes, then*

$$s_\mu s_\nu = \sum_{\lambda} c_{\mu\nu}^\lambda s_\lambda.$$

There's a proof with Knuth-equivalence and concatenation.

We know that  $s_\nu \cdot s_\mu$  can be obtained by finding  $s_T$  where  $T$  is the skew shape formed by concatenating right down of  $\mu$  with LEFT UP  $\nu$ . Then the product is

$$\sum_{\lambda} c_{R\lambda}^\rho s_\lambda.$$

## 1. SYMMETRIC FUNCTIONS

---

To prove it we use Fulton's techniques [3].

We want to show  $c_{R\lambda}^\rho = c_{\mu\nu}^\lambda$ . This means that the number of Yamanouchi tableaux of shape  $\rho/R$  and content  $\lambda$  is the same as  $\text{sh}(\lambda/\mu)$  and content  $\nu$ .

*Remark 1.15.10.* Consider

3	4				
2	2	3			
1	1	2	4		
				3	
				2	2
				1	1

the lower one has to be  $T_\nu$  in any Yamanouchi filling.

We first de-RSK it to get

$$\begin{array}{|c|c|c|c|c|} \hline 3 & 3 & 3 & & \\ \hline 2 & 2 & 2 & 2 & \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline \end{array} \rightarrow \begin{pmatrix} 123331222111 \\ 111112222333 \end{pmatrix}$$

if  $r$  is the first row and  $b$  the second one, then we insert  $r$  into  $U$ . But when inserting we are keeping track with  $b$ , so we label the new squares with  $b$  as a skew recording tableau.

Name  $Q$  such tableau, we claim that  $Q$  is a ballot SSYT shape  $\lambda/\mu$  content  $\nu$ .

Putting this together in the example we get

$$U = \begin{array}{|c|c|c|c|} \hline 3 & 4 & & \\ \hline 2 & 2 & 3 & \\ \hline 1 & 1 & 2 & 4 \\ \hline \end{array} \leftarrow 123331222111$$

READ IN FULTON AND UNDERSTAND BIJECTION

## Chapter 2

# Representation Theory: A Crash Course

### 2.1 Day 15 | 20230222

Representation is a big and not unified area, so we will talk only about what we need for combinatorics.

**Definition 2.1.1.** A group is a set  $(G, *)$  such that  $*$  is a binary operation which is associative, possesses an identity element and every element possesses an inverse.

We say a matrix representation of a group is an assignment  $g \mapsto M_g \in \text{GL}_n(\mathbb{C})$  for  $g \in G$  with  $M_g M_h = M_{g*h}$ .

If the assignment is injective we say the representation is faithful.

**Example 2.1.2.** Some groups that we will use are  $(S_n, \circ)$ , the general linear group  $(\text{GL}_n(\mathbb{C}), \cdot)$ , the Multiplicative subgroup of the real numbers  $(\mathbb{R} \setminus \{0\}, \cdot)$ .

On the other hand group representations of  $S_3$  we have that the composition is the operation. We can trivially represent it as  $\sigma \mapsto (1)$  for every  $\sigma \in S_3$ . This representation is not faithful.

We also have the *sign representation*

$$\begin{array}{lll} \diamond \text{id} \mapsto (1). & \diamond (13) \mapsto (-1). & \diamond (123) \mapsto (1). \\ \diamond (12) \mapsto (-1). & \diamond (23) \mapsto (-1). & \diamond (132) \mapsto (1). \end{array}$$

These are the only 1-dimensional representations of  $S_3$ . In fact for  $S_n$  this is true, there are only 2 one-dimensional representation.

We can represent our permutations as permutation matrices!

## 2. REPRESENTATION THEORY: A CRASH COURSE

---

$$\begin{aligned}
 \diamond \text{id} &\mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, & \diamond (13) &\mapsto \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, & \diamond (123) &\mapsto \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \\
 \diamond (12) &\mapsto \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, & \diamond (23) &\mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, & \diamond (132) &\mapsto \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}.
 \end{aligned}$$

In general  $\sigma \mapsto M_\sigma$  where  $(M_\sigma)_{i,j} = \delta_{i,\pi(j)}$ . This representation is faithful, but it's *too big*.

We can notice that  $v = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$  is an eigenvector with eigenvalue 1 of all of these matrices. The eigenspace  $\text{gen}(v)$  is fixed by all of these matrices. The orthogonal complement to  $\text{gen}(v)$  is rotated about the normal axis when acted by this matrices, such plane is

$$\text{gen} \left( \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} \right).$$

If we add our eigenvector to the basis then

$$[\text{id}]_{\mathcal{B}}^{\mathcal{C}} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & -1 & 1 \\ 1 & 0 & -1 \end{pmatrix} \Rightarrow [\text{id}]_{\mathcal{C}}^{\mathcal{B}} = \frac{1}{3} \begin{pmatrix} 1 & 1 & 1 \\ 2 & -1 & -1 \\ 1 & 1 & -2 \end{pmatrix}.$$

**Lemma 2.1.3.** *If  $g \mapsto M_g$  is a representation, then  $g \mapsto A^{-1}M_gA$  is also a representation.*

**Proof**

$$A^{-1}M_gAA^{-1}M_hA = A^{-1}M_{g*h}A.$$

Conjugating with our matrix we get the following representation:

$$\begin{aligned}
 \diamond \text{id} &\mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, & \diamond (12) &\mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 1 \end{pmatrix}, & \diamond (13) &\mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}.
 \end{aligned}$$

The fourth representation of  $S_3$  comes from the two-by-two components of the block diagonal matrices in question. Knowing where the generators go is sufficient to

know where every element maps. This means that

$$(132) = (12)(13) \mapsto \begin{pmatrix} -1 & 1 \\ -1 & 0 \end{pmatrix}.$$

$$(123) = (13)(12) \mapsto \begin{pmatrix} 0 & -1 \\ 1 & -1 \end{pmatrix}.$$

$$(23) = (13)(132) \mapsto \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}.$$

But how would we have found this representation without recurring to the basis? We can think of our representation as we did, every group element is a matrix.

Another way is a homomorphism  $G \rightarrow \text{GL}_n(\mathbb{C})$ . Also we can consider it as a vector space space action

$$G \cdot V : G \times V \rightarrow V.$$

With this understanding 3 and 3' aren't fundamentally different.

**Definition 2.1.4.** Two representations  $g \mapsto M_g$  and  $g \mapsto N_g$  are isomorphic whenever there exists  $A \in \text{GL}_n(\mathbb{C})$  such that  $A^{-1}M_gA = N_g$ .

In terms of the vector spaces, two representations  $V, W$  are isomorphic if there's an isomorphism that preserves the group action.

- ◇ Can we have isomorphic representations between dimensions? No, but,
- ◇ can we have non-isomorphic representations of the same dimension? Yes! The one dimensional ones aren't isomorphic.

## Operations on representations

The direct sum acts on vector spaces as  $V \oplus W$  or on maps as

$$(g \mapsto M_g) \oplus (g \mapsto N_g) = g \mapsto \begin{pmatrix} M_g & \\ & N_g \end{pmatrix}.$$

**Example 2.1.5.** The permutation representation for  $S_3$  is the direct sum of the standard one and the trivial one.

**Definition 2.1.6.** A representation is irreducible<sup>1</sup> if it is not the direct sum of smaller dimensional representation.

---

<sup>1</sup>indecomposable means the same as irreducible when working over  $\mathbb{C}$ .

Any number is a unique product of primes, so in an analogous way, any representation is a direct sum of irreducibles.

**Theorem 2.1.7** (Maschke/Schur). *Every complex representation of a finite group  $G$  is uniquely (up to isomorphism) a direct sum of irreducibles.*

We will see that irreducible representations of the symmetric group correspond to Schur functions.

### 2.2 Day 16 | 20230224

The main goal of a representation is to break it down into irreducible representations. We want to classify into irreducible representations.

**Proposition 2.2.1.** *The number of irreducible representations of a finite group  $G$  is the number of conjugacy classes.*

Recall  $g, h \in G$  are conjugate if  $h = kgk^{-1}$  for some  $k \in G$ .

**Example 2.2.2.** If  $G$  is abelian, the conjugacy classes are singletons. This means that the number of irreducible representations of  $G$  is  $|G|$ . As we represent over complex numbers, we map each element of  $G$  to a  $|G|^{\text{th}}$  root of unity.

**Example 2.2.3.** Recall that conjugacy classes in the symmetric group are determined by cycle type. Notice that the number of conjugacy classes is  $5 = p(4)$ .

Partition	Cycle type	Elements of conjugacy class
(1, 1, 1, 1)	4 1-cycles	() (identity element)
(2, 1, 1)	1 transposition	(12), (13), (14), (23), (24), (34)
(2, 2)	2 transpositions	(12)(34), (13)(24), (14)(23)
(3, 1)	1 3-cycle	(123), (132), (234), (243), (341), (314), (412), (421)
(4)	1 4-cycle	(1234), (1243), (1324), (1342), (1423), (1432)

Symmetric Group  $S_4$  cycle structure

We now have 2 irreducible representations, the trivial and sign one. The rest we will find with tableaux.

**Proposition 2.2.4.** *There is one irreducible representation  $V_\lambda$  of  $S_n$  for each partition  $\lambda \vdash n$ .*



### Specht's construction of $V_\lambda$

Sometimes this  $V_\lambda$ 's are called Specht modules, this means irreducible representation of  $S_n$ . First remember that  $S_n$  acts on  $\mathbb{C}[x_1, x_2, \dots, x_n]$ .

We can think of  $(\mathbb{C}[x_1, \dots, x_n], +)$  as an infinite dimensional  $S_n$  representation. This in the sense that

$$\mathbb{C}[x_1, x_2, \dots, x_n] = \bigoplus_{d=0}^{\infty} \{ \text{degree } d \text{ homogenous component} \}$$

We will construct  $V_\lambda$  as a sub-representation of  $\mathbb{C}[x_1, \dots, x_n]$ .

**Definition 2.2.5.** Let  $T$  be a standard filling (1 to  $n$  no restrictions on order) of a shape  $\lambda$ . The Garnir polynomial is

$$F_T = \prod_{(*)} (x_j - x_i)$$

where we sum through  $i$  "below"  $j$  in the same column of  $T$ .

**Example 2.2.6.** Consider the partition  $(2, 2, 1)$  and a filling

4	
1	3
5	2

so we look at the first column. We begin by adding factors

$$(x_4 - x_1)(x_4 - x_5)(x_1 - x_5)$$

and in the second column, only get  $(x_3 - x_2)$ . So  $F_T$  is the product

$$F_T = (x_4 - x_1)(x_4 - x_5)(x_1 - x_5)(x_3 - x_2).$$

**Definition 2.2.7.** The Specht module is  $V_\lambda = \text{gen}\{ F_T : T \text{ is a standard filling of } \lambda \}$ .

**Lemma 2.2.8.**  $S_n$  acts on  $V_\lambda$  by permuting variables.

**Example 2.2.9.** In the previous example,  $(12)$  acts on  $F_T$  as

$$(12) \cdot F_T = (x_4 - x_2)(x_4 - x_5)(x_2 - x_5)(x_3 - x_1)$$

so  $(12)T$  is

4	
2	3
5	1

## 2. REPRESENTATION THEORY: A CRASH COURSE

The proof of the lemma 2.2.8 begins by determining the action of  $\pi \in S_n$  over the basis. We can see that  $\pi F_T = F_{(\pi T)}$  and after that and we extend linearly to all elements, since the action on the generators determines the action on the whole space.

**Proposition 2.2.10.** *The set of  $F_T$ 's such that  $T$  is a SYT is a basis for  $V_\lambda$ . In particular*

$$\dim(V_\lambda) = \#SYT(\lambda)$$

where we can use the hook-length formula.

We will see the proof next class, for now, an example:

**Example 2.2.11.** Consider  $V_{(2,1)}$ , we have the following possible fillings of  $\text{sh}(2, 1)$ :

$$\begin{array}{|c|} \hline 3 \\ \hline 1 \ 2 \\ \hline \end{array}, \quad \begin{array}{|c|} \hline 2 \\ \hline 1 \ 3 \\ \hline \end{array}, \quad \begin{array}{|c|} \hline 1 \\ \hline 2 \ 3 \\ \hline \end{array}, \quad \begin{array}{|c|} \hline 3 \\ \hline 2 \ 1 \\ \hline \end{array}, \quad \begin{array}{|c|} \hline 2 \\ \hline 3 \ 1 \\ \hline \end{array}, \quad \begin{array}{|c|} \hline 1 \\ \hline 3 \ 2 \\ \hline \end{array}.$$

By taking the Garnir polynomial of each filling  $T$  we can find the generators for our  $V_{(2,1)}$ :

$$V_{(2,1)} = \text{gen}(x_3 - x_1, x_2 - x_1, x_1 - x_2, x_3 - x_2, x_2 - x_3, x_1 - x_3)$$

Clearly we don't need some of them due to signs. After eliminating the negatives, we can see that

$$x_3 - x_2 = (x_3 - x_1) - (x_2 - x_1)$$

so that we may eliminate it as well. We are only left with  $x_3 - x_1$  and  $x_2 - x_1$ , the ones that correspond to SYT and we see that the dimension of  $V_{(2,1)}$  is indeed 2, the number of SYT's of shape  $(2, 1)$ .

**Exercise 2.2.12.** Compute  $V_{(1,1,1)}$  and  $V_{(3)}$ . What is their dimension? [Hint: We saw both of this representations on last class.]

### Answer

Both representations have dimension 1 as there's only one SYT of shape  $(1, 1, 1)$  and  $(3)$ :

$$\begin{array}{|c|} \hline 3 \\ \hline 2 \\ \hline 1 \\ \hline \end{array}, \quad \text{and} \quad \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline \end{array}.$$

For  $V_{(3)}$  this is the span of 1, because we can't make any Garnir polynomials. Also  $S_3$  acts on trivially on 1. This means  $V_{(3)}$  that is the trivial representation. For  $V_{(1,1,1)}$  we have that it's generated by  $(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)$ . We can see

that this is the sign representation.

Let's now see why the representation  $V_{(2,1)}$  is the same as the standard representation we found last time.

First,  $V_{(2,1)}$  is a subspace of  $\mathbb{C}[x, y, z]_{(1)}$ , the homogenous linear polynomials of degree 1 and we have

$$\mathbb{C}[x, y, z]_{(1)} = \text{gen}(x, y, z).$$

As  $V_{(2,1)} = \text{gen}(z - x, y - x) = \text{gen} \left( \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \right)$ . This is the action of  $S_3$  on the orthogonal subspace  $\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}^\top$ .

**Lemma 2.2.13.** *The permutation representation of  $S_n$  is*

$$V_{(n)} \oplus V_{(n-1,1)}.$$

On monday we will continue with tensor products and  $V_\lambda \rightarrow s_\lambda, V_\lambda \otimes V_\mu \rightarrow s_\lambda s_\mu u$

## 2.3 Day 17 | 20230227

Going back to Yamanouchi words on 2 letters, the way to solve it is to consider the crystals. Each connected component contains a sequence with three 2's and two 1's which can be counted using  $\binom{n}{\lfloor n/2 \rfloor}$  as this question is equivalent to asking number of words of length  $2n + 1$  with  $n + 1$  ones and  $n$  twos.

### Garnir polynomials

Recall that  $T$  is a filling on  $\lambda$  with  $[n]$  with no restrictions on rows or columns.  $T$ 's Garnir polynomial is  $\prod (x_i - x_j)$  where  $i$  is "above"  $j$ . From this,  $V_\lambda$  is  $S_n$ -invariant. This means that  $\pi v \in V_\lambda$  when  $v \in V_\lambda$  and  $\pi \in S_n$ .

We now need a *straightening algorithm* to convert  $F_T$ 's to other  $F_T$ 's.

**Lemma 2.3.1** (Column straightening). *If  $T$  is a filling of  $\lambda$  with  $[n]$  and  $T'$  is formed from  $T$  by ordering columns, then  $F_{T'} = (\pm)F_T$ .*

## 2. REPRESENTATION THEORY: A CRASH COURSE

---

**Example 2.3.2.** Consider the fillings  $T$  and  $T'$  respectively:

$$\begin{array}{|c|c|c|c|} \hline 8 & 7 & & \\ \hline 1 & 2 & 5 & \\ \hline 4 & 9 & 3 & 6 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|} \hline 8 & 9 & & \\ \hline 4 & 7 & 5 & \\ \hline 1 & 2 & 3 & 6 \\ \hline \end{array}.$$

We have

$$\begin{cases} F_T = (x_8 - x_1) \dots \\ F_{T'} = (x_8 - x_1) \dots \end{cases}$$

Proof was omitted in class **LOOK AT MARIA NOTES**

**Lemma 2.3.3** (Row straightening). Suppose  $T$  is column-increasing and we do the following:

- i) Choose the topmost row having a decrease (if it's not row increasing).
- ii) Choose the rightmost decrease in that row.
- iii) Define the Garnir operator

$$g_{A,B} = \sum_{(*)} \text{sgn}(\sigma) \sigma$$

where the sum is taken over  $\sigma \in S_{A \cup B}$  which preserve column increases.

Then

$$g_{A,B} F_T = 0 \iff \sum_{\sigma} \text{sgn}(\sigma) F_{\sigma(T)} = 0.$$

**Example 2.3.4.** In the filling

$$\begin{array}{|c|c|c|c|} \hline 8 & 9 & & \\ \hline 4 & 7 & 5 & \\ \hline 1 & 2 & 3 & 6 \\ \hline \end{array}$$

we have  $A$  to be the column above and including the 7, while  $B$  is the column including and below 5. Then the Garnir operator is

$$g_{A,B} = \text{id} - (57) + (37)(59) + (735) + (597) - (3597).$$

The result of applying each term of the operator to the filling results in

$$\begin{array}{|c|c|c|c|} \hline 8 & 9 & & \\ \hline 4 & 5 & 7 & \\ \hline 1 & 2 & 3 & 6 \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 8 & 5 & & \\ \hline 4 & 3 & 9 & \\ \hline 1 & 2 & 7 & 6 \\ \hline \end{array}, \text{quad} \begin{array}{|c|c|c|c|} \hline 8 & 9 & & \\ \hline 4 & 3 & 7 & \\ \hline 1 & 2 & 5 & 6 \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 8 & 7 & & \\ \hline 4 & 5 & 9 & \\ \hline 1 & 2 & 3 & 6 \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 8 & 7 & & \\ \hline 4 & 3 & 9 & \\ \hline 1 & 2 & 5 & 6 \\ \hline \end{array}.$$

and notice that (property is now there, which one column inc or row incr?) so that

$$F_T = F_1 - F_2 - F_3 - F_4 + F_5$$

**Proposition 2.3.5.** Irreducibility of  $V_{\lambda}$

### Tensor product on representations

The *inner* tensor product is defined on all representations whereas the *outer* tensor product is specific to the  $S_n$  case.

**Proposition 2.3.6.** *If  $V, W$  are vector spaces of dimension  $n$  and  $m$ , then  $\dim(V \otimes W) = mn$ .*

The construction of the tensor product that we will use is the one of the free vector space over  $V \times W$  modulo the relations of bilinearity.

**Proposition 2.3.7.** *If  $V, W$  are  $S_n$ -representations, then  $V \otimes W$  is an  $S_n$  representation as well given by*

$$\pi(v \otimes w) = (M_\pi v \otimes M_\pi w).$$

The Kronecker problem is that we have  $V_\lambda$  and  $V_\mu$  and we would like to determine the decomposition of  $V_\lambda \otimes V_\mu$  into irreducibles.

## 2.4 Day 18 | 20230301

The inner tensor product is

$$g(v \otimes w) = M_g v \otimes N_g w$$

so if  $V, W$  represent,  $V \otimes W$  represents. For matrices  $A \otimes B$  is a block matrix.

### Outer Tensor Product

**Definition 2.4.1.** Suppose  $H \leq G$  for a group  $G$  and  $V$  represents  $H$ . Then the induced representation of  $G$  is

$$\mathbb{C}G \otimes_{\mathbb{C}H} V.$$

A basis is the basis of  $V$  tensored by coset elements.

**Example 2.4.2.** Consider the trivial representation of  $S_2$ , let us call it  $1_{S_2}$ . Let us now induce from  $S_2$  to  $S_3$ , we are fixing the 3 so  $S_2 = \text{gen}((12))$ . Thus we may find a basis

$$v_1 = 123 \otimes v_0, \quad v_2 = 132 \otimes v_0, \quad v_3 = 312 \otimes v_0,$$

where  $v_0$  is the only basis element of  $1_{S_2}$ . Sending this to matrices we get the permutation representation.

**Definition 2.4.3.** If  $V$  represents  $G$  and  $H \leq G$ , then restricting the representation to  $H$  means just taking the *ones* in  $H$ .

With the induced and restricted representations we can now go on to defining the outer tensor product of representations.

**Definition 2.4.4.** The outer tensor product of  $V, W$  representations of  $S_n$  and  $S_m$  is

$$Ind_{S_n \times S_m}^{S_{n+m}} V \otimes W.$$

Notice that  $V \otimes W$  is a representation of  $S_n \times S_m$  so we induce it to the larger group  $S_{n+m}$ . Notice that we may interpret a permutation in  $S_n \times S_m$  as a permutation of the word

The difference is that  $V, W$  in the inner one are in the same group. Now we get an  $S_{n+m}$  representation from the outer. The Kronecker problem looks for irreducible representations of  $V_\lambda \otimes_{in} V_\mu = \bigoplus g) \lambda \mu^\nu V_\nu$ .

However it is possible to find the decomposition of the outer tensor product:

$$V_\mu \otimes_{out} V_\nu = \bigoplus_\lambda c_{\mu\nu}^\lambda V_\lambda$$

where  $c_{\mu\nu}^\lambda$  is the Littlewood-Richardson coefficient. This is analogous to the product of two Schur functions

$$s_\mu s_\lambda = \sum_\lambda c_{\mu\nu}^\lambda s_\lambda.$$

**Example 2.4.5.** Consider  $V_{(2)} \otimes_{out} V_{(1)}$ , this is the trivial representation of  $S_2 \times S_1$  which means this is the induction from  $S_2 \times S_1$  to  $S_3$ .

Notice that the group  $S_2 \times S_1 \simeq S_3$ , so from before, we found that this was the permutation representation of  $S_3$ . We have decomposed this (REF WHERE) as

$$V_{(2,1)} \oplus V_{(3)}.$$

This is analogous to  $s_{(2)} s_{(1)}$ .

**Also read example in notes column sign, row trivial** We wish to understand the previous correspondence.

**Definition 2.4.6.** The Frobenius map takes a representation  $V = \bigoplus c_\lambda V_\lambda$  to  $\sum c_\lambda s_\lambda$ . In other words, takes irreducibles  $V_\lambda$  to basis elements  $s_\lambda$  and extend by linearity.

Notice that

$$\text{Frob}(V_\lambda \otimes_o V_\mu) = s_\lambda s_\mu \Rightarrow \text{Frob}(V \otimes_o W) = \text{Frob}(V) \text{Frob}(W).$$

We may extend to virtual representations which are complex formal sums of representations. Then  $\text{Frob} : (\text{Virt}, \oplus, \otimes) \rightarrow \Lambda$  is a ring homomorphism.

But what's the deal with Schur positivity, this theory gives us a corollary.

**Corollary 2.4.7.** *A homogenous degree  $d$  symmetric function is a Frobenius character of some representation if and only if  $f$  is Schur positive.*

This means that there are two strategies when working in combinatorial representation theory.

- (a) We have a representation for which we can decompose  $\text{Frob}(V)$  into Schur functions, then we can find a decomposition of  $V$  into irreducibles.
- (b) If we have a collection of symmetric functions which seem to be Schur positive, then we can show that the fact is true by finding a representation  $V_i$  such that  $f_i = \text{Frob}(V_i)$ .

## 2.5 Day 19 | 20230303

**Exercise 2.5.1.** Write  $s_{(2,1)}$  in terms of  $p_\lambda$  and  $p_{(2,1)}$  in terms of  $s_\lambda$ .

Recall that we have some formulas to go between basis of symmetric functions. The full is available in Maria's webpage. With what we know, we can't solve the previous exercise.

### Murnaghan-Nakayama Rule

**Definition 2.5.2.** The character  $\chi_V$  of a representation  $V$  of a group  $G$  is

$$\chi_V : G \rightarrow \mathbb{C}, g \mapsto \text{tr}(M_g) = \sum(\text{diagonal elements}) = \sum(\text{eigenvalues}).$$

No matter what basis we choose, the character is invariant.

**Lemma 2.5.3.**

$\chi_V$  is constant on conjugacy classes.

**Proof**

$$\chi_V(hgh^{-1}) = \text{tr}(M_{hgh^{-1}}) = \text{tr}(M_h M_g M_h^{-1}) = \text{tr}(M_g) = \chi_V(g).$$

From this we can think of the character as a function from the conjugacy classes. As the amount of irreducible representations is the number of conjugacy classes, then we can write a character table

TABLE

Characters are also additive across representations, this means that

$$\chi_{V \oplus W} = \chi_V + \chi_W.$$

So knowing irreducibles allows us to find characters for all other representations. The other direction is much more important, if we know  $\chi_V$ , we can uniquely decompose it as a linear combination of irreducible characters. So we can deduce the decomposition of  $V$  into irreducible representations.

**Example 2.5.4.** We will find the character table of  $S_2$ . As  $S_2$  is abelian, every element is in its own conjugacy class:

$S_2$	id	(2)
$\chi(2)$	1	1
$\chi(1,1)$	1	-1

Character table for  $S_2$

**Example 2.5.5.** We will find the character table of  $S_3$ :

$S_3$	id	(12)	(123)
$\chi(3)$	1	1	1
$\chi(1,1,1)$	1	-1	1
$\chi(2,1)$	2	0	-1

Character table for  $S_3$

The character table for  $\chi_{\text{reg} S_3}$  is (600) so we may decompose the character as

$$2(20 - 1) + (111) + (1 - 11) \Rightarrow V_{\text{reg}} = \text{triv} \oplus \text{sign} \oplus \text{std}^2.$$

The regular representation always decomposes a *dim \* character*. This is an application of RSK as well **AUDIO 20min**

We will alternatively define the Frobenius map with characters.

**Definition 2.5.6.** Suppose  $V$  represents  $S_n$ , then

$$\text{Frob}(V) = \frac{1}{n!} \sum_{\pi \in S_n} \chi_V(\pi) p_{c(\pi)}$$

where  $c((12)(35)(467)) = (3, 2, 2)$ .



This is summing over  $n!$  elements, but remember some of them share cycle type.

**Lemma 2.5.7.** *Under this definition, this can also be written as*

$$\text{Frob}(V) = \sum_{\lambda \vdash n} \chi_V(\pi \in \lambda \text{conj.class}) \frac{p_\lambda}{z_\lambda}$$

where  $z_\lambda = \prod k^{m_k} m_k!$ .

**Proof**

We have that the size of the conjugacy class of cycle type  $\lambda$  is  $n!/z_\lambda$ . So rewriting the Frobenius map we get

$$\text{Frob} = \frac{1}{n!} \sum_{\pi \in S_n} \chi_V(\pi) p_{c(\pi)} = \frac{1}{n!} \sum_{\lambda \vdash n} \sum_{\pi \text{ cyc.type } \lambda} \chi_V(\pi) p_\lambda$$

Since character is constant, we have

$$\frac{1}{n!} \sum_{\lambda \vdash n} (\# \text{conj.class cycledtype } \lambda) \chi_V(\pi_\lambda) p_\lambda$$

that amount is  $n!/z_\lambda$  so factorials cancel out and we get the desired formula.

**Example 2.5.8.** Recall that the character table for  $\chi_{(2,1)}$  is  $(2, 0, -1)$  so

$$\text{Frob}(V_{(2,1)}) = 2 \frac{p_{(1,1,1)}}{z_{(1,1,1)}} + 0 p_{(2,1)}/z - \frac{p_3}{z_3} = \frac{1}{3} p_1^3 - \frac{1}{3} p_3 = m_{(2,1)} + 2m_{(1,1,1)} = s_{(2,1)}.$$

Now for instance from the trivial character,  $\text{Frob}(V_{(3)})$ , the character table is  $(1, -1, 1)$  and

In essence the coefficients are the character tables.

**Theorem 2.5.9.** *The character tables for  $S_n$  are the transition matrices from  $(s_\lambda)$  to the normalized power sum:  $(p_\lambda/z_\lambda)$ .*

The product for this fact can be found in [10]. This theorem is also equivalent to the fact that both Frobenius maps are equivalent.

We now ask,

*Is there a combinatorial way to compute  $\chi_\mu(\lambda)$ ?*

To derive it, we will use a duality. The Schur and power sum bases are self dual with respect to the Hall inner product so

$$\chi_\mu(\lambda) = \text{coeff.} \frac{p_\lambda}{z_\lambda} \quad \text{in} \quad s_\mu = \langle s_\mu | p_\lambda \rangle = \text{coeff.} s_\mu \quad \text{in} \quad p_\lambda.$$

So  $p_\lambda = \sum_\mu \chi_\mu(\lambda) s_\mu$ .

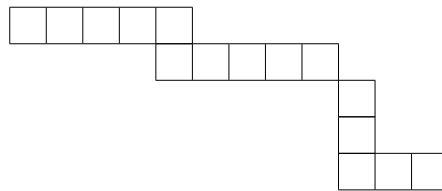
### 2.6 Day 20 | 20230306

#### The Inverse Murnaghan-Nakayama Rule

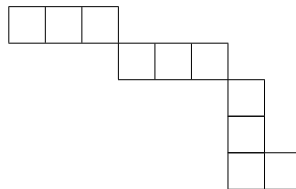
We wish to compute  $\chi_U(\lambda)$ , the character of  $V_\mu$  on a permutation  $\pi$  of cycle type  $\lambda$ . This is the coefficient  $\langle s_\mu | p_\lambda \rangle$ , the coefficient of  $s_\mu$  in  $p_\lambda$ .

**Definition 2.6.1.** A border strip or a ribbon is a connected skew-shape containing no  $2 \times 2$  square.

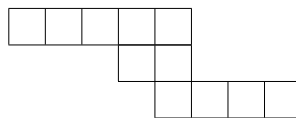
**Example 2.6.2.** The tableau



is a ribbon, the tableau



is not a ribbon because it's not connected. Finally the tableau



has a  $2 \times 2$  square.

**Definition 2.6.3.** The height of a border strip  $\lambda/\mu$  is  $\# \text{rows} - 1$ .

A border strip tableau with shape  $\lambda$  is a filling with positive integers such that

- ◇ row and columns are weakly increasing
- ◇ Every integer forms a border strip.

**Example 2.6.4.** The tableau

3	3	5	5	
1	2	4	5	
1	2	2	2	2
1	1	1	1	2

is a border strip tableau

**Theorem 2.6.5.** *The character*

$$\chi_{\mu}(\lambda) = \langle s_{\mu} | p_{\lambda} \rangle = \sum_{(*)} \prod_{i=1}^{\ell(\lambda)} (-1)^{\text{ht}(\text{strip}(i))}$$

where the sum runs through border strip tableaux  $T$  with shape  $\mu$  and content  $\lambda$ .

**Example 2.6.6.** Let us find  $\langle s_{\text{sh}(3,2,1)} | p_{(3,3)} \rangle$ . We wish to fill border strip tableau with 111222. We have the following fillings

2						1					
2	2					1	2				
1	1	1				1	2	2			

Calculating the heights we have

$$(-1)^0(-1)^1 = -1, \quad \text{and} \quad (-1)^2(-1)^2 = -1$$

so  $\langle s_{\text{sh}(3,2,1)} | p_{(3,3)} \rangle = -1 - 1 = -2$ . The coefficients might not be positive but at least they are integers.

**Exercise 2.6.7.** Calculate  $\langle s_{\text{sh}(2,1)} | p_{(2,1)} \rangle$ .

The tableaux are

2				1			
1	1			1	2		

so we get  $1 - 1 = 0$ .

## 2. REPRESENTATION THEORY: A CRASH COURSE

**Example 2.6.8.** Notice that  $\langle s_{\text{sh}(3,2)} | p_{(1,1,1,1,1)} \rangle$  is the number of standard Young tableaux because we are filling with 12345.

On the other hand

$$\langle s_{\text{sh}(\mu)} | p_{(n)} \rangle$$

is either 0 if  $\mu$  is not a *hook* and  $(-1)^{\ell(\mu)-1}$  when it actually is a hook.

The following proof comes from Egge's symmetric function, we will separate it into stages.

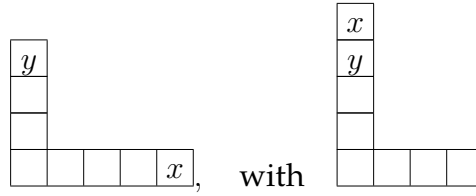
**Lemma 2.6.9.**  $p_r = \sum_{j=0}^{r-1} (-1)^j s_{(r-j,1,1,\dots,1)}$  where there are  $k$  ones.

**Proof**

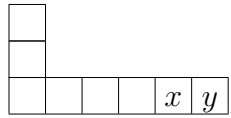
We have that

$$s_{\text{hook}} = \sum_{(*)} x^T$$

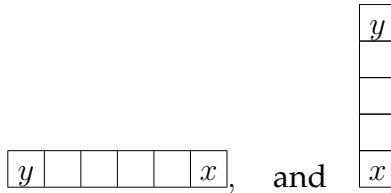
we wish to cancel  $x^T$ 's as follows



if  $x > y$ . Else if  $x \leq y$  we pair with



This is a sign-reversing involution and the only edge cases are



so the fixed points are tableau of the form



which contribute monomials of the form  $x_i^r$ . So monomials that don't cancel out are the power sum symmetric functions.

Let's see an example

**Example 2.6.10.** What about  $p_4$ ?

**Lemma 2.6.11.** We have

$$s_\mu p_r = \sum_{(*)} (-1)^{ht(\lambda/\mu)} s_\lambda$$

where the sum runs through  $\lambda$  such that  $\lambda/\mu$  is a border strip of size  $r$ .

Recall the Pieri rules which say

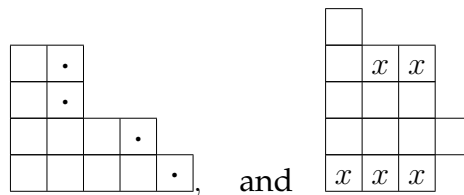
$$s_\mu h_r = \sum_{(*)} s_\lambda$$

where the sum is through  $\lambda$  such that  $\lambda/\mu$  is a horizontal strip of size  $r$ . The  $e$  Pieri rule instead goes

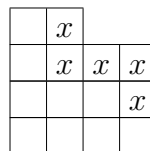
$$s_\mu e_r = \sum_{(*)} s_\lambda$$

where the sum now is through vertical strips of size  $r$ .

The tableaux



represent the vertical and horizontal strips. But the tableau



represents the border strip.

### Proof

First notice that

$$s_{(r-j, 1, 1, \dots, 1)} = \sum_{k=0}^j (-1)^{j-k} h_{r-k} e_k.$$

The idea is that hook shapes are similar to vertical and horizontal strips.

**Example 2.6.12.** Consider  $s_{\text{sh}(3,1,1)}$ . By the formula this should be

$$h_{\text{sh}(5)}e_0 - h_4e_1 + h_3e_2$$

and we stop here because the height here is  $j = 2$ . This is the same as

$$s_{\text{sh}(5)}e_0 - s_4e_1 + s_3e_2.$$

By the Pieri rule this is

$$s_{\text{sh}(5)} - (s_{\text{sh}(5)} + s_{\text{sh}(4,1)}) + (s_{\text{sh}(4,1)} + s_{\text{sh}(3,1,1)}).$$

So all terms cancel except for  $s_{(r-j, 1, 1, \dots, 1)}$ .

Substituting this result in the lemma

$$s_\mu p_r = \sum_{i=0}^{r-1} (-1)^j s_\mu \sum_{k=0}^j (-1)^{k-j} h_{r-k} e_k.$$

The  $(-1)$  gets a  $k$  power and we are left with

$$\sum_{j=0}^{r-1} \sum_{k=0}^j (-1)^k s_\mu h_{r-k} e_k$$

Where  $h$  adds horizontal strips and  $e$  adds vertical strips. And they are gonna cancel unless they are a border strip.

$r$	$r$	$b$				
		$b$				
				$r$	$b$	
						$r$

So the difference is that looking at the shape, this contributes  $s_\lambda$ . We can *toggle something* so it does change the height.

# Chapter 3

## Counting with Group Actions

### 3.1 Day 21 | 20230308

**Definition 3.1.1.** An action of a group  $G$  on a set  $X$  is a map

$$G \times X \rightarrow X, (g, x) \mapsto g \cdot x$$

such that  $h \cdot (g \cdot x) = (hg) \cdot x$  and  $\text{id} \cdot x = x$ .

We have already seen group actions when  $X$  is a vector space. Representations are actions:  $M_g \in \text{End}(V)$ . This map is bijective because we can find an inverse to the action.

**Example 3.1.2.** A group action which is not a representation is  $S_n$  acting on  $[n]$  by  $\pi \cdot i = \pi(i)$ . This is an action because composition does what we want it to do.

Also  $S_n$  acts on itself:

- ◇ By conjugation:  $\pi \cdot \sigma = \pi \sigma \pi^{-1}$ .
- ◇ By left-multiplication:  $\pi \cdot \sigma = \pi \sigma$ .

Every group acts on itself by multiplication.

**Definition 3.1.3.** An action  $G \cdot X$  is transitive if  $g \cdot x = y$  for some  $g \in G$  and all  $x, y \in X$ .

On the previous examples,  $S_n \cdot [n]$  is transitive. Left multiplication is transitive, but conjugation is not.

**Example 3.1.4.** The group  $\mathbb{Z}/5\mathbb{Z}$  can be thought of  $\langle (12345) \rangle \leq S_5$ . This acts on the set of points of a 5-pointed star by rotation.

### 3. COUNTING WITH GROUP ACTIONS

---

We wish to study the orbits of the actions which tell us how *not-transitive* an action is.

**Definition 3.1.5.** The orbit of a point  $x \in X$  under an action  $G \cdot X$  is

$$\text{Orb}(x) = \{ g \cdot x : g \in G \}.$$

The idea is that when we have a cyclic group, these sets look like *orbits around the center*. Also, be quick to notice that

$$\text{Orb}(x) = \text{Orb}(g \cdot x), \quad g \in G$$

which means that the orbits partition the set  $X$ . In particular, “being in the same orbit” is an equivalence relation.

In the previous examples:

- ◇ Conjugations’s orbits are the conjugacy classes.
- ◇ Left multiplication’s orbit is only one.

**Example 3.1.6.** For the action  $(S_3 \times S_2) \cdot [5]$ , we have orbits  $[3]$  and  $\{4, 5\}$ .

Our objective now is to count orbits. This count is related to stabilizers.

**Definition 3.1.7.** The stabilizer of  $x \in X$  under  $G \cdot X$  is

$$\text{Stab}(x) = \{ g \in G : g \cdot x = x \}.$$

**Example 3.1.8.** In  $S_n \cdot [n]$  we have

$$\text{Stab}_{S_n}(n) = S_{n-1}$$

because we can freely move the other  $n - 1$  elements. Now consider the conjugation action on  $S_5$ , then

$$\text{Stab}_{S_5}((123)(45)) = C_3 \times C_2.$$

Now how do the orbit and stabilizer relate? The main reference is [8] in chapter 6.

**Theorem 3.1.9 (Orbit-Stabilizer Theorem).** For  $G \cdot X$  a group action, and a point  $x \in X$  we have

$$|\text{Stab}(x)| |\text{Orb}(x)| = |G|.$$



**Proof**

If  $s$  is the size of the stabilizer, then we consider  $y \in \text{Orb}(x)$ . This means that  $y = g \cdot x$ , so we get  $s$  elements which send  $x$  to  $y$ . In other words

$$h \cdot x = g \cdot x \Rightarrow (g^{-1}h) \cdot x = x$$

so  $g^{-1}h \in \text{Stab}(x)$ , i.e.  $h = g \cdot r$  for some  $r \in \text{Stab}(x)$ .

So there exists  $s$  elements of  $G$  sending  $x$  to  $y$  for  $y \in \text{Orb}(x)$ .

**Example 3.1.10.** We will find the size of a conjugacy class of cycle type  $\lambda$ . We can identify this as the orbit of the conjugacy action. This is, by Orbit-Stabilizer:

$$\frac{|S_n|}{|\text{Stab}(\pi_\lambda)|}.$$

For example if  $\lambda = (3, 3, 2, 1, 1)$  then  $\pi_\lambda = (123)(456)(78)(9)(10)$ . To stabilize this we can still do cyclic group actions on the cycles. This accounts for  $C_3 \times C_3 \times C_2$ . We add two  $2!$  reswitching. If  $m_i$  is the number of  $i$ 's in  $\lambda$ , then  $m_1 = 2$ ,  $m_2 = 1$  and  $m_3 = 2$ . So we have:

$$\frac{|S_n|}{|\text{Stab}(\pi_\lambda)|} = \frac{n!}{\prod_i i^{m_i} m_i!} = \frac{n!}{z_\lambda}.$$

We can also find the size of the group with this.

**Example 3.1.11.** We ask, how many rotation in 3D fix a cube? If we fix  $x$  to be a corner of the cube, then

$$|\text{Stab}(x)| = 3, \quad |\text{Orb}(x)| = 8,$$

then we have 24 elements in our group.

We could also fix an edge to get 2 and 12 or fixing faces 4 and 6.

**Burnside's Lemma**

*It's actually not Burnside but Frobenius' lemma.* With this result we will count orbits. The question now is *in how many ways can we 2-color the faces of a cube up to rotation?* It's 10, not 8. **What did I overcount?**

**Lemma 3.1.12.** *The number of orbits of an action is*

$$\frac{1}{|G|} \sum_{g \in G} |\text{Fix}(g)|$$

where  $\text{Fix}(g) = \{x \in X : g \cdot x = x\}$ .

### 3. COUNTING WITH GROUP ACTIONS

---

To solve the previous question we have that the key is the phrase “up to rotation”. A coloring up to rotation is an orbit under  $F$  of 2-colorings of a labeled cube. There are  $2^6$  such colorings. Now we have that the number of orbits is

$$\frac{1}{24} (2^6 + 8 \cdot 2 + 6 \cdot 2^3 + 6 \cdot 2^3 + 3 \cdot 2^4)$$

identity, 120, 90 deg through face, 6 edge rotations, and the last one is through the face.

#### 3.2 Day 22 | 20230310

##### Orbit Counting

We have asked in how many ways we can 2-color faces of a cube up-to-rotation.

FIGURE

Alternatively with Burnside’s lemma, if  $G$  is the rotation group of a cube and  $X$  is the set of 2-colorings of faces of *labeled* cubes, then the number we are looking for is the number of orbits of  $G \cdot X$ . The elements of  $G$  are

- ◇ identity
- ◇ 8 120 vtx
- ◇ 6 +-90 throu faces
- ◇ 3 180 thru faces
- ◇ 6 180 thru opp edges

Then adding

$$\frac{2^6 + 8 \cdot 2^3 + 6 \cdot 2^3 + 3 \cdot 2^4 + 6 \cdot 2^3}{24} = \frac{2^3 + 4 + 6 + 6 + 6}{3} = 10.$$

The proof of Burnside’s lemma is a double counting argument.

##### Proof

We will count in two ways the set

$$\{ (g, x) : g \cdot x = x \}.$$

For each  $g$ , there's  $|\text{Fix}(g)|$  pairs  $(g, x)$  such that  $g \cdot x = x$ . So counting through all  $g \in G$ ,

$$|\{(g, x) : g \cdot x = x\}| = \sum_{g \in G} |\text{Fix}(g)|.$$

Now for  $x \in X$ , there are  $|\text{Stab}(x)|$   $g$ 's such that  $g \cdot x = x$ , then

$$|\{(g, x) : g \cdot x = x\}| = \sum_{x \in X} |\text{Stab}(x)|.$$

The sum of the stabilizers can be interpreted as

$$\sum_{x \in X} |\text{Stab}(x)| = \sum_{x \in X} \frac{|G|}{|\text{Orb}(x)|} = |G| \sum_{x \in X} \frac{1}{|\text{Orb}(x)|}.$$

We may break the sum as

$$\sum_{x \in X} \frac{1}{|\text{Orb}(x)|} = \sum_{O \text{ orbit}} \sum_{x \in O} \frac{1}{|O|} = \sum_{O \text{ orbit}} \frac{|O|}{|O|} = \#(\text{orbits}).$$

This proves the result.

**Example 3.2.1.** Consider the action  $S_3 \times S_2 \cdot [5]$ . In this case

$$\sum_{x \in [5]} |\text{Stab}(x)| = \sum_{x \in [5]} \frac{6}{|\text{Orb}(x)|} = 6 \sum_{x=1}^5 \frac{1}{|\text{Orb}(x)|} = 6 \left( \frac{1}{3} + \frac{1}{3} + \frac{1}{3} + \frac{1}{2} + \frac{1}{2} \right).$$

**Example 3.2.2.** Consider the action of  $S_n$  on binary sequences of length  $n$ . The number of orbits is  $n + 1$ , because orbits are determined by amount of ones of a string.

By Burnside's lemma

$$\# \text{orbits} = \frac{1}{n!} \sum_{\pi \in S_n} |\text{Fix}(\pi)|$$

By considering **something ASK** this sum becomes

$$\frac{1}{n!} \sum_{\lambda \vdash n} (\# \pi'_\lambda s) 2^{\ell(\lambda)}$$

and we may conclude

$$n + 1 = \sum_{\lambda \vdash n} \frac{2^{\ell(\lambda)}}{z_l a}.$$

### 3. COUNTING WITH GROUP ACTIONS

---

We may also derive this from row-shapes! We have

$$s_{\text{row}} = \sum \frac{p_\lambda}{z_\lambda} \Rightarrow s_{\text{row}}(x, y) = \sum \frac{p_\lambda(x, y)}{z_\lambda}$$

This begs us to ask the question, how many SSYT have only ones and twos with shape  $(n)$ ? This is  $s_{(n)}(1, 1) = n + 1$ . So we get

$$n + 1 = \sum_{\lambda \vdash n} \frac{p_\lambda(1, 1)}{z_\lambda} = \sum_{\lambda \vdash n} \frac{2^{\ell(\lambda)}}{z_\lambda}.$$

Examples like this can be worked with Polya theory. Look at [8].

#### Cyclic Sieving Phenomenon

This topic was discovered in the early 2000's, it can be read in Sagan's [8]. The big idea is summing roots of unity, we will say that  $\omega_n = e^{\frac{2\pi i}{n}}$ . If  $\omega = \omega_n$  then

$$1 - \omega^n = (1 - \omega)(1 + \omega + \cdots + \omega^{n-1}) = 0 \Rightarrow 1 + \omega + \cdots + \omega^{n-1} = 0.$$

**Definition 3.2.3.** A triple  $(X, G, f)$  exhibits the cyclic sieving phenomenon if  $G \cdot X$  and  $f$  is a polynomial with integer coefficients such that

$$|\text{Fix}(g)| = f(\omega_{\text{ord}(g)}), \quad \text{where} \quad \text{ord}(g) = \min\{k : g^k = \text{id}\}.$$

Many times, the polynomials  $f(q)$  are  $q$ -analogues.

**Example 3.2.4.** Consider  $G = C_n \leq S_n$  generated by  $(123 \dots n)$  acting on  $X = \left(\left(\begin{smallmatrix} [n] \\ k \end{smallmatrix}\right)\right)$ , this is the set of all multisets of  $k$  chosen from  $[n]$ .

If  $f(q) = \left(\begin{smallmatrix} n \\ k \end{smallmatrix}\right)_q$ , then this makes a triple  $(X, G, f)$  exhibiting the cyclic sieving phenomenon.

For  $n = 4$  and  $k = 2$ , then

$$X = \{11, 12, 13, 14, 22, 23, 24, 33, 34, 44\} \Rightarrow |X| = \left(\left(\begin{smallmatrix} 4 \\ 2 \end{smallmatrix}\right)\right) = \binom{5}{2} = 10.$$

Now  $f(q) = \frac{(5)_q(4)_q}{(2)_q}$ , simplifying we obtain

$$(1 + q + q^2 + q^3 + q^4)(1 + q^2)$$

Then plugging in  $q = i$  we get zero, which coincides with  $|\text{Fix}(1234)|$ . In the case of  $(13)(24)$ , this only fixes 13 and 24 also  $f(-1) = 2$  as  $-1$  is a second root of unity.

Now notice that when taking the identity, every element is fixed and  $f(1) = 10$ .

# Index

action, 63  
antisymmetric, 8  
border strip, 58  
border strip tableau, 58  
character, 55  
corner, 31  
dual, 14  
forgotten basis, 16  
Frobenius map, 54  
Garnir operator, 52  
Garnir polynomial, 49  
Hall inner product, 14  
highest weight, 40  
horizontal strip, 24  
inner corner, 31  
inner slide, 32  
irreducible, 47  
Knuth move, 27  
Knuth-equivalent, 27  
Kronecker problem, 53, 54  
lowering operator, 38

orbit, 64  
orthogonal, 14  
orthonormal, 14  
outer corner, 31  
outer tensor product, 54  
Plactic monoid, 35  
raising operator, 38  
reading word, 21, 33  
rectification, 32, 34  
Schur polynomial, 10  
Schur-positive, 8  
skew shape, 24  
skew-SSYT, 31  
stabilizer, 64  
standarization, 21  
straight, 32  
symmetric, 5  
two-line array, 17  
Vandermonde determinant, 10  
vertical strip, 24  
Yamanouchi, 36



# Bibliography

- [1] François Bergeron, Gilbert Labelle, and Gilbert Leroux. *Combinatorial Species and Tree-like Structures*. Cambridge University Press, 1998.
- [2] Peter Jephson Cameron and Jacobus Hendricus van Lint. *Designs, Graphs, Codes and their Links*. London Mathematical Society Student Texts. Cambridge University Press, 1991.
- [3] William Fulton. *Young Tableaux: With Applications to Representation Theory and Geometry*. London Mathematical Society Student Texts #35. Cambridge University Press, 1999.
- [4] Grayson Graham. The ring of symmetric polynomials. *REUsChicago*, dunno.
- [5] Ronald Lewis Graham, Donald Ervin Knuth, and Oren Patashnik. *Concrete mathematics: a foundation for computer science*. Addison-Wesley, 2nd ed edition, 1994.
- [6] James Oxley. *Matroid Theory*. Oxford University Press, USA, 2nd edition, 2011.
- [7] Bruce Eli Sagan. *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*. Graduate Texts in Mathematics №203. Springer, 2 edition, 2001.
- [8] Bruce Eli Sagan. *Combinatorics: The Art of Counting*. Graduate Studies in Mathematics. American Mathematical Society, 2020.
- [9] Richard Peter Stanley. *Enumerative Combinatorics: Volume 1*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2011.
- [10] Richard Peter Stanley and Sergey Fomin. *Enumerative Combinatorics: Volume 2*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1997.