18.217 — Young Tableaux

Class by Alex Postnikov

Notes by Sanjana Das

Fall 2022

Notes for the MIT class **18.217** (Young Tableaux), taught by Alex Postnikov. All errors are my responsibility.

Contents

§1 September 7, 2022

This class will be about Young tableaux — it won't be *only* about Young tableaux, but they will be one of the main players. There are several recommended (but not required) textbooks — if you're serious about combinatorics, it's a good idea to have them:

- Richard Stanley's *Enumerative Combinatorics, Volume 2* (last year this course concentrated on Volume 1), especially Chapter 7 (where Young tableaux appear a lot).
- William Fulton's *Young Tableaux.*
- Bruce Sagan's *The Symmetric Group.*

These books have some intersection, but they present similar constructions from several points of view. In this class, we'll look at the same object from different perspectives.

Definition 1.1. A **partition** $\lambda = (\lambda_1, \dots, \lambda_k)$ of n is a nondecreasing sequence of integers with $\lambda_1 + \cdots + \lambda_k = n$.

Definition 1.2. A **Young diagram** is the associated figure to a partition where we have λ_i boxes in the ith row.

For example, $\lambda = (3, 2, 2) \vdash 7$ corresponds to

$$\ydiagram{3,2,2}\,.$$

A Young tableaux is then a way of writing numbers in a Young diagram, in certain ways.

Definition 1.3. A **standard Young tableau** (SYT) is a way of writing 1, 2, 3, ..., n in the boxes of a Young diagram (without repetition) so that they increase in both rows and columns.

Example 1.4

One Young tableaux for $(3, 2, 2)$ is

$$\begin{ytableau} 1 & 2 & 4 \\ 3 & 6 \\ 5 & 7 \end{ytableau}\,.$$

We'll later discuss *semistandard Young tableaux* (where we allow repetition).

In combinatorics, you can define lots of combinatorial structures, but not all these structures will be interesting. We want to concentrate on *interesting* objects. What's interesting may be subjective, but Prof. Postnikov's criteria is that they appear in many different areas of math. From this point of view, SYT are very interesting — they appear in several areas of math.

Young tableaux show up in the representation theory of GL_n, SL_n, and S_n — when trying to understand this representation theory on a combinatorial level, these structures appear. (Representation theory also relates to crystals.)

They're also closely related to *symmetric functions* (polynomials invariant when you permute variables).

They're *also* related to geometry — more specifically, Schubert calculus (Grassmanians, flag manifolds, ...). (This also relates to Schubert polynomials, and various other natural generalizations.)

This is why we have three different textbooks — Stanley's textbook mostly focuses on symmetric functions, Fulton's on geometry, and Sagan's on representations of S_n. Of course, these perspectives are closely related — you can go from representation theory to symmetric functions by taking characters.

Another point of view from representation theory is that it's natural to identify Young diagrams with Gelfand-? patterns, which provides a polytope point of view — Young tableaux can be identified with lattice points of polytopes. In particular, there are various bijections (for example, Robinson-Schensted-Knuth) which can be understood from the point of view of polytopes as well — you can cut one polytope into pieces and recombine them to get another. This can be called *piecewise linear combinatorics*, which in turn is closely related to tropical geometry. You can then detropoicalize and get into the realm of birational combinatorics, cluster algebras, and so on.

These words may or may not appear in this class, but there is a natural chain of generalizations leading from Young tableaux to these topics.

§2 Standard Young Tableaux

§2.1 Catalan Numbers

Example 2.1

Consider the partition $\lambda = (n, n)$, corresponding to the Young diagram

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

How many SYT are there?

Theorem 2.2

The number of SYT of shape (n, n) is the nth **Catalan number**

$$C_n = \frac{1}{n+1}\binom{2n}{n}.$$

There are *many* different combinatorial interpretations of Catalan numbers (Richard Stanley has a list of more than 200 interpretations). But one of the most common interpretations is the number of *Dyck paths*. SYT of shape (n, n) are in an easy bijection with Dyck paths. We read numbers one by one, and if they're in the first row we take an up-step $(1, 1)$, and if they're in the second we take a down-step $(1, -1)$.

Example 2.3

When $n = 5$, the tableau

$$T = \begin{array}{|c|c|c|c|c|}\hline 1 & 2 & 4 & 7 & 9 \\ \hline 3 & 5 & 6 & 8 & 10 \\ \hline \end{array}$$

corresponds to the Dyck path $++-+--+-+-$.

This gives a path from $(0, 0)$ to $(2n, 0)$ that stays weakly above the x-axis — at every stage, the number of numbers in the first row is at least the number in the second, so the path stays weakly above the x-axis.

There's many proofs that the number of Dyck paths is $\frac{1}{n+1}\binom{2n}{n}$. There's a cute proof using the reflection method, and another using cyclic shifts. This is Prof. Postnikov's favorite method.

Proof. Rewrite the formula as
$$\frac{1}{n+1}\binom{2n}{n} = \frac{1}{2n+1}\binom{2n+1}{n}.$$

This suggests a combinatorial interpretation — $\binom{2n+1}{n}$ is the number of *all* lattice paths (paths with up and down steps, but without the condition that they stay above the x-axis) with n up-steps and $n+1$ down-steps.

Example 2.4

For $n = 5$, we should have 5 up-steps and 6 down-steps, for example $+---++-++--$.

So these are paths from $(0,0)$ to $(2n+1,-1)$.

Claim — Among all $\binom{2n+1}{n}$ arbitrary lattice paths, exactly $\frac{1}{2n+1}$ of them stay above the x-axis until their last step.

The last step is always below the x-axis, but if that's the only place, we can call these *almost Dyck paths* — they're Dyck paths with an extra down-step. And we claim the probability that a randomly chosen path is an almost Dyck path is $\frac{1}{2n+1}$.

Example 2.5

The path $+-++---+--$ is an almost Dyck path.

Proof. We want to group these $\binom{2n+1}{n}$ paths into groups of size $2n+1$, where exactly one element of each group is an almost Dyck path.

Example 2.6

Take $n = 3$, with initial path $+--++--$. Then we express this path by a sequence of $+$'s and $-$'s, as $+--++--$. Now we take the *cyclic shifts* of this sequence, where we take the first entry and move it to the end — this gives

$$+--++--, --++--+, -++--+-, ++--+--, +--+--+, --+--++, -+--++-.$$

We always have $2n+1$ cyclic shifts. This gives a bunch of groups.

Claim — Each group has $2n+1$ elements — in other words, there are no repetitions among the cyclic shifts.

Proof. We have $\gcd(n, n+1) = 1$. If there are repetitions (after d cyclic shifts we get the same thing) then the sequence has to be periodic in a nontrivial way — it's the same sequence repeated k times. But then $\gcd(n, n+1)$ is divisible by k, contradiction. ∎

Claim — Exactly one of these cyclic shifts is good.

In our example, the good one is the fourth $(++--+--)$.

Proof. Take the example $+-+---+-++--+++--$. When we perform cyclic shifts, we take the first step and attach it to the end. So we're cutting the path at some point, writing p as the concatenation of $p' \circ p''$, and reversing their order to $p'' \circ p'$.

We claim that we need to cut at the *first minimal point.* It's possible to show that this works, while if we cut in any other place then the path goes below the x-axis. ■

Later when we talk about crystals, finding the first minimal point of something will be an important construction. ■

□

Definition 2.7. For a partition λ, f^λ is the number of SYT of shape λ.

In this language, we've proven that $f^{(n,n)} = C_n$. But it turns out that for *any* partition λ, there's an explicit product formula for the number of SYT, called the **Hook-Length Formula**.

So from this point of view, we can think of SYT as a generalization fo Dyck path, and the hook-length formula as a generalization fo the formula for Catalan numbers. We'll discuss the Hook-Length Formula at some point. There's many proofs; a *nice* one is the probabilistic hook walk. (It will not be repeated here because it was given last semester in 18.212.) THere is another nice proof based on the polyhedral point of view — there's a way to interpret both sides of the identity as volumes of certain polytopes, and in order to prove the formula we construct a piecewise linear volume-preserving transformation between the polytopes (we cut them into pieces and recombine them). This gives the Hook-length Formula but also more general results. This will require more theory but we will probably see it at some point.

In the near future, we'll see that these objects naturally appear in *computer science*. The book by Donald Knuth which is a sort of bible of computer science has a third volume about sorting, and from this point of view it's very natural to arrive to combinatorics that we're going to discuss. In fact, a large part of this book discusses SYT and semistandard Young tableaux, and in particular the RSK correspondence (which will be one of the main topics in this class).

Suppose we have a permutaiton of $[n]$, for example $2, 3, 5, 1, 6, 7, 4$, and we want to sort them. One way to do so is to use a **queue**, which is first-in first-out (FIFO) — you can place entries into the queue, and you extract in the same order you place.

If you have one queue, you cannot change the numbers — they'll come out in the same order they came in. But now suppose you have *two* queues. Then you can read the numbers one by one and place them in the first or second queue, and you can get them out from either queue as well. Your goal is to sort the permutation.

This example is sortable (2, 3, 5 in the first, 1 in the second, get 1 out, get 2 and 3 out, put 6,)

Definition 2.8. A **queue-sortable permutation** is a permutation which is sortable using two queues.

Theorem 2.9

The number of queue-sortable permutations of $[n]$ is C_n.

With one queue you can't sort anything, with two you can sort C_n. But how many can be sorted with k queues? In general there's no nice answer but there's an asymptotic one, found using techniques we will discuss in this class.

§3 September 9, 2022

Last class, we saw that $f^{(n,n)} = C_n$ has a nice formula. This formula can be generalized to one for *any* Young diagram:

Definition 3.1. Given a partition $\lambda \vdash n$, for each box $(i,j) \in \lambda$, we define its **hook length** h_{ij} as the number of boxes below it and to its right, including itself.

Theorem 3.2 (Frome–Robinson–Thrall, Hook Length Formula)

We have

$$f^\lambda = \frac{n!}{\prod_{(i,j)\in\lambda}} h_{ij}.$$

For a $2 \times n$ rectangle, it's easy to see this is equivalent to the formula for the Catalan numbers.

Example 3.3

Take the partition

$$\lambda = (2,3) = \begin{array}{|c|c|c|}\hline & & \\ \hline & \\ \cline{1-2}\end{array}.$$

Then the hook lengths are

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

So we have

$$f^{(3,2)} = \frac{5!}{1 \cdot 1 \cdot 2 \cdot 3 \cdot 4} = 5.$$

The original proof of the Hook Length Formula is quite complicated, so people were trying to find nicer proofs. There are some nicer proofs — for example, there's a probabilistic proof based on a random walk on a Young diagram. Today we will present another proof, which Prof. Postnikov thinks is the best proof — the easiest to understand and most conceptual. The nice thing is that once you understand the essence of this proof, you automatically get lots of other constructions for free — RSK, the octahedral recurrence, cluster algebras, and so on. So in some sense, half of mathematics is based on the same core principle. This construction appears under many different names in different areas, but it's the same elementary building block.

§3.1 Polytopal Proof

Fix a partition λ, and define two convex polytopes in $\mathbb{R}^n$.

The first polytope is

$$\Delta_\lambda := \{(x_{ij})_{(i,j)\in\lambda} \mid x_{ij} \geq 0, \sum h_{ij}x_{ij} \leq 1\}.$$

Our other polytope is

$$P_\lambda := \{(y_{ij})_{(i,j)\in\lambda} \mid y_{ij} \geq 0, y_{ij} \leq y_{i,j+1}, y_{ij} \leq y_{i+1,j}, \sum y_{ij} \leq 1\}.$$

So we essentially take our Young diagram and fill it with entries (this is a sort of generalized version of a matrix)

$$\begin{array}{|c|c|}\hline x_{11} & x_{12} \\ \hline x_{21} \\ \cline{1-1}\end{array}$$

Example 3.4

Take $\lambda = (3, 2)$. Then

$$\Delta_\lambda = \{\begin{array}{|c|c|c|}\hline x_{11} & x_{12} & x_{13} \\ \hline x_{21} & x_{23} & \\ \cline{1-2}\end{array} \mid x_{ij} \geq 0, 4x_{11} + 3x_{12} + x_{13} + 2x_{21} + x_{22} \leq 1\}.$$

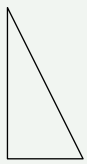
Meanwhile

$$P_\lambda = \{\begin{array}{|c|c|}\hline y_{11} & \cdots \\ \hline\end{array} \mid y_{ij} \geq 0, y_{11} \leq y_{12} \leq y_{23}, y_{11} \leq y_{21}, \dots, \sum y_{ij} \leq 1\}.$$

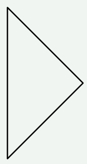
This is a five-dimensional polytope, so we can't draw it on a blackboard. Instead we'll take a more trivial example so that we can draw it:

Example 3.5

Take $\lambda = (2)$. Then Δ_λ is $x_1, x_2 \geq 0$ and $2x_1 + x_2 \leq 1$.



Meanwhile for the second, we have $0 \leq y_1 \leq y_2$ and $y_1 + y_2 \leq 1$.



Note that both polytopes have area $\frac{1}{4}$.

There's a simple map sending our first polytope to the second: $(x_1, x_2) \mapsto (x_1, x_1 + x_2)$. We would like to generalize this to any λ.

The reason Δ is called Δ is that it is a simplex for any λ — it's the standard simplex with rescaled coordinates. This means

$$\mathrm{Vol}(\Delta_\lambda) = \frac{1}{\prod h_{ij}} \mathrm{Vol}\left((x_{ij}) \mid x_{ij} \geq 0, \sum x_{ij} \leq 1\right)$$

by setting $x'_{ij} = h_{ij} x_{ij}$, so then

$$\mathrm{Vol}(\Delta_\lambda) = \frac{1}{n!} \cdot \frac{1}{\prod_{(i,j) \in \lambda} h_{ij}}.$$

Now let's calculate $\mathrm{Vol}(P_\lambda)$. We have an array of y_{ij} which are weakly decreasing in rows and columns. All possible orderings of the y's correspond to a SYT, so then

$$\mathrm{Vol}(P_\lambda) = f^\lambda \,\mathrm{Vol}\{(y_1, \dots, y_n) \mid 0 \leq y_1 \leq y_2 \leq \cdots \leq y_n, y_1 + \cdots + y_n \leq 1\}.$$

We can write this as

$$f^\lambda \cdot \frac{1}{n!} \mathrm{Vol}((y_{ij}) \mid y_i \geq 0, \sum y_i \leq 1),$$

since there are $n!$ possible orderings in the standard coordinate simplex and each gives a piece with the same volume. So then this polytope has volume

$$\frac{1}{(n!)^2} \cdot f^\lambda.$$

So the Hook-Length formula says that the first volume equals the second!

So far we've just reformulated the hook-length formula in this polytopal form — it's equivalent to showing that $\operatorname{Vol} \Delta_\lambda = \operatorname{Vol} P_\lambda$.

The best way to see that two polytopes have the same volume is to construct a volume-preserving map between them.

Theorem 3.6

There exists a continuous bijective piecewise-linear volume-preserving map $\varphi_\lambda \colon \Delta_\lambda \to P_\lambda$.

In our example we found a *linear* map. In general you may not be able to — the first polytope is a simplex but the second might not be — but there is a *piecewise linear* map, which means we break Δ into various polytope pieces and define a linear map on each of these pieces. These linear maps should be volume-preserving, meaning that they're given by a matrix whose determinant is 1.

Student Question. Why do we care about continuous?

We don't need it to show the volume equality, but it turns out this stronger version is true, and implies many other things than HLF.

In fact, we're not just going to show the map exists, we're going to construct it.

We will actually construct a map between *cones*

$$\varphi_\lambda \colon \{(x_{ij})_{(i,j)\in\lambda} \mid x_{ij} \geq 0\} \to \{(y_{ij})_{(i,j)\in\lambda} \mid y_i \geq 0, \text{weakly decreasing}\}$$

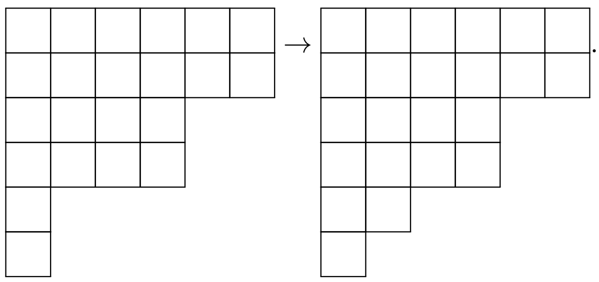
such that

$$\sum h_{ij} x_{ij} = \sum y_{ij}.$$

Then requiring the first expression is at most 1 is the same as requiring the second is, which means the two finite pieces of the cones we care about are sent to each other.

Proof. We construct the map by induction on $|\lambda|$. The base case is when $\lambda = \varnothing$ is the empty partition of 0. Then both polytopes are zero-dimensional and just consist of a single point, so $\Delta_\varnothing = P_\varnothing = \{\cdot\}$.

Now for the inductive step, assume we already know how to construct φ_λ, and we want to construct the map for a slightly larger Young diagram μ — where we add one extra box.



Now we want a map

$$\varphi_u : \text{array of } x\text{'s} \to \text{array of } y\text{'s}.$$

It turns out we don't have to modify many entries of the y_{ij} (in the output of our map) — we only have to modify the y_{ij} on the diagonal where we added the box.

Let's denote the entries in this diagonal by $y_1, y_2, \ldots$. We also need to know the diagonals immediately below and above this one: call the diagonal below $y'_1, \ldots$, and the one above as $y''_1, y''_2, \ldots$. The whole operation will only depend on these three diagonals, and we will modify only the y_i. Rotate the picture so that our diagonal is now horizontal: so we have

$$\begin{array}{cccccc} y'_1 & & y'_2 & & y'_3 & \\ & y_1 & & y_2 & & y_3 \\ y''_1 & & y''_2 & & y''_3. & \end{array}$$

Then we have some inequalities: these correspond to

$$y_i \in [\max(y'_{i+1}, y''_{i+1}), \min(y'_i, y''_i)].$$

So we can think of this as a line segment, and y_i is somewhere in this interval.

We would like to do some transformation to y_i such that after thsi operation, it's still inside this interval,a nd we want our map to be bijective — so this operation has to be invertible. The easiest thing we can do is flip the interval — reflect y_i with respect to the midpoint of this interval, to get a new value y^*_i. In coordinates,

$$y^*_i = \min(y'_i, y''_i) + \max(y'_{i+1}, y''_{i+1}) - y_i.$$

This is called a **toggle move**.

So now we have a map φ_μ sending our slightly extended array to the new array where we put $y^*_1, \ldots$ instead of $y_1, \ldots$, and we also add one extra array y^*_0. We need to have $y^*_0 \geq \max(y'_1, y''_1)$, so we take

$$y^*_0 = \max(y'_1, y''_1) + z$$

where z is the new entry.

If we try to use something outside the shape, we assume that it's 0.

Claim — This works.

We can easily check all needed properties, but first let's seen an example.

Example 3.7

Start with $\varphi_{(1)}$. The map is just $\boxed{x_{11}} \mapsto \boxed{x_{11}}$.

For $\varphi_{(2)}$ we start with this map and add one extra box. We don't want to change x_{11}, and the diagonal is empty, so our map becomes

$$\begin{array}{|c|c|}\hline x_{11} & x_{12} \\ \hline\end{array} \mapsto \begin{array}{|c|c|}\hline x_{11} & x_{11}+x_{12} \\ \hline\end{array}.$$

Then for $\varphi_{(2,1)}$, we start with the above thing, and we don't modify anything because the diagonla of the bottom thing is empty. So we get

$$\begin{array}{|c|c|}\hline x_{11} & x_{12} \\ \hline x_{21} \\ \cline{1-1}\end{array} \mapsto \begin{array}{|c|c|}\hline x_{11} & x_{12} \\ \hline x_{11}+x_{21} \\ \cline{1-1}\end{array}.$$

It's easy to see that this becomes weakly decreasing. In fact, for any hook shape, the map will be linear.

Now consider $\varphi_{(2,2)}$ (obtained from adding a box to the previous one). We only modify the top-left corner, so we need to toggle it. So we get

$$\begin{array}{|c|c|}\hline x_{11} & x_{12} \\ \hline x_{21} & x_{22} \\ \hline\end{array} \mapsto \begin{array}{|c|c|}\hline \min(x_{11}+x_{12}, x_{11}+x_{21}) - x_{11} & x_{11}+x_{12} \\ \hline x_{11}+x_{21} & \max(x_{11}+x_{12}, x_{11}+x_{21}) + x_{22} \\ \hline\end{array}$$

We can simplify this a bit by factoring out x_{11} in both places.

We want to see that the sum of the y's is the same as the sum of x's scaled by hook-lengths. (It's easy to see invertible, since toggle is invertible; it's easy to see it's piecewise linear because max is, and you can easily check that all maps involved are volume-preserving because toggles are volume-preserving.) The least trivial property is the sum of the two entries.

Note that $\min(a, b) + \max(a, b) = a + b$. So in our last example, we get

$$\sum y_{ij} = 3x_{11} + 2x_{21} + 2x_{12} + x_{22}.$$

So the coefficients in front of the x's are exactly the hook lengths, and we claim that this is true in general. □

Student Question. If you do this inductive processes in different ways do you get the same map?

The construction is based on how you add boxes but it doesn't actually depend on the boxes, only on the shape. The reason is if you want to add a box in one place and then another, if there's two possible ways you can go, if these two operations commute — you can add box a and then box b, or b and then a — then a and b cannot be in adjacent diagonals so they do not influence each other. So then you can use hte DiamonD Lemma argument and deduce that the map is independent of the order in which you add boxes.

§4 September 12, 2022

Recall that we constructed a map φ_λ from the array of x's to y's inductively, by supposing we add a box with new entry z. Then we also add a box on the right-hand side, and the entry we put in that box is $\max(\mu, \delta) + z$ where μ, δ are the entries left and above that box in the original y_{ij}. Simultaneously, we also *toggle* all entries on the diagonal in which we added the box: where toggling means we send

$$\begin{array}{|c|c|c|} \cline{2-2} \multicolumn{1}{c|}{} & a & \multicolumn{1}{c}{} \\ \hline b & y & c \\ \hline \multicolumn{1}{c|}{} & d & \multicolumn{1}{c}{} \\ \cline{2-2} \end{array} \to \begin{array}{|c|c|c|} \cline{2-2} \multicolumn{1}{c|}{} & a & \multicolumn{1}{c}{} \\ \hline b & y^* & c \\ \hline \multicolumn{1}{c|}{} & d & \multicolumn{1}{c}{} \\ \cline{2-2} \end{array}$$

where $y^* = \max(a, b) + \min(c, d) - y$.

Example 4.1

For $(2, 2)$ we get

$$\begin{array}{|c|c|} \hline x & y \\ \hline z & t \\ \hline \end{array} \mapsto \begin{array}{|c|c|} \hline \min(y,z) & x+y \\ \hline x+z & x+t+\max(y,z) \\ \hline \end{array}$$

It's easy to see that this is bijective (all steps are reversible) and piecewise linear. It's also volume-preserving — you can check that if you pick either value of the max, you get a map with determinant 1. But we also finally need to show that it sends the relevant simplexes to each other.

Lemma 4.2

If φ_λ sends (x_{ij}) to (y_{ij}), then

$$\sum h_{ij} x_{ij} = \sum y_{ij}.$$

In fact, there is a stronger lemma: there are *many* different linear relations between the x's and y's. To do so, we'll define a few things.

First, concentrate on the border ribbon of the Young diagram (the boxes at the bottom and right). Number these boxes of the ribbon 1 to ℓ from left to right. These numbers tell us the labels of diagonals in the Young diagram. For every box k, we consider the kth rectangle R_k which has bottom-right corner k, and goes up and left as far as possible. We also look at the diagonal starting at this box, which we call D_k.

Lemma 4.3

For each k, $\sum_{(i,j) \in R_k} x_{ij} = \sum_{(i,j) \in D_k} y_{ij}$.

So rectangular sums of the x_{ij} equal diagonal sums of y_{ij}. Denote these sums by r and d.

Example 4.4

In the $(2, 2)$ case, $r_1 = x + z$, $r_2 = x + y + z + t$, and $r_3 = x + y$. Meanwhile $d_1 = x + z$,

$$d_2 = \min(y, z) + \max(y, z) + x + t = x + y + z + t,$$

and finally $d_3 = x + y$.

First let's see why Lemma 2 implies Lemma 1.

Proof of 2 $\implies$ 1. We have $r_1, \dots, r_\ell$ the rectangular sums of x_{ij}. Meanwhile $d_1, \dots, d_\ell$ are the rectangular sums of y_{ij}. We know $\sum d_i = \sum y_{ij}$, since every entry appears in exactly one diagonal. Meanwhile $\sum r_i = \sum ? x_{ij}$ for some multiplicities.

$\square$

For the part of the ribbon that starts in the ith column from the jth column to the ith row, any box there will have rectangle containing x_{ij}. So the multiplicity of x_{ij} is the length of this ribbon. But that's exactly the hook length — if you draw the hook, you can see that these lengths are the same. $\square$

So that's why hook lengths appear — they appear as sizes of sub-ribbons in our border ribbon.

Now it remains to prove Lemma 2. The proof is actually pretty simple.

Proof of Lemma 2. We use induction. Suppose we have φ_λ with $r_1, \ldots, r_\ell$ and $d_1, \ldots, d_\ell$, and now we add one extra box. When we do this, the rectangular sums may change, and the diagonal sums may change. Only one rectangular sum changes — if our corner is at the mth position, only r_m changes. Similarly only d_m changes. All the other values remain the same, so they remain equal; so it suffices to check that the ones which change change in the same way.

The rectangular sum adds z, as well as a layer to the left and above (since we get a bigger rectangle). So that's the $(m-1)$st rectangle.

(Note that all our dots' positions remain the same, except the mth dot moves down and to the right one.)

So then $r_m \mapsto \tilde{r}_m = r_{m-1} + r_{m+1} - r_m + z$ (we subtract r_m because we've overcounted the intersection).

Meanwhile on the other side, only one value changes — only d_m changes. The formula involved diagonals $a_1, a_2, \ldots$ in the $(m-1)$th diagonal, $b_1, b_2, b_3, \ldots$ in the mth, and $c_1, c_2, c_3, \ldots$ in the $(m+1)$th. We have $d_{m-1} = a_1 + \cdots$, $d_m = b_1 + \cdots$, and $d_{m+1} = c_1 + c_2 + \cdots$. When we add a box, we have $d_m \rightsquigarrow \widetilde{d}_m$ where

$$\widetilde{d}_m = (\max(a_1, c_1) + z) + (\max(a_2, c_2) + \min(a_1, c_1) - b_1) + (\max(a_3, c_3) + min(a_2, c_2) - b_2) + \cdots.$$

Now we can see that we have $\max(a_1, c_1) + \min(a_1, c_1) = a_1 + c_1$, and so on. So we can replace all the max and min with the ordinary sums of the entries. This gives

$$(a_1 + a_2 + \cdots) + (c_1 + c_2 + \cdots) - (b_1 + b_2 + \cdots) + z.$$

This is exactly what we need! So $\widetilde{d}_m = \tilde{d}_{m-1} + \tilde{d}_{m+1} - d_m + z$. This means we get the exact same rule for diagonal and rectangular sums, so they remain the same. $\square$

That's the end of the proof of HLF, but that's not the end of the story — we proved not only the identity that we need, but many other identities as well. SO we can try to prove some claims stronger than the HLF. In fact, this construction implies many different things — it actually implies more than half of the theorems we will prove in this class.

Remark 4.5. Logistical things:

- Canvas page will exist at somepoint, but we will use it only for problem sets.
- The first problem set will be assigned in a week or two, once we have enough material for a problem set.
- There are a lot of diagrams. If Prof, Postnikov tries to prepare lecture notes drawing all diagrams, it would take forever. But some students are good at this. If some of us have great skills like this, we should tell him. Otherwise, he was teaching remote classes for some time, and prepared notes for his remote classes. We are not really following any previous classes, but some of the material is contained in lectures from past years; he can put links to these, or to relevant sections of textbooks.
- How many problem sets will there be? 3 ± 1. Usually there are 12 problems but you do not need to solve all of them; there will be a minimal requirement for an A (such as 5/12). PSets will appear on the course webpage. Canvas will just be used to submit. He will also not tell us which ones are easy and hard, but some are.

This will be on the problem set:

Exercise 4.6. Suppose that λ is a $m \times n$ rectangle with $m \leq n$. Then we want to find SYT of this rectangular shape. Now look at the main diagonal — the entries on the main diagonal are $a_1 < a_2 < \cdots < a_m$. Now we assign the *weight* of T as

$$\frac{1}{\prod_{i=1}^{m} i^{a_{i+1}-a_i}}.$$

Note that a_{m+1} is by convention defined to be $a_{mn}+1$. Prove that

$$\sum_T \mathrm{wt}(T) = 1.$$

(The sum is over all SYT of this shape.)

Student Question. Is something like this true for other shapes as well?

There are many formulas — there's a stronger claim in general.

Example 4.7

For a 2×3 rectangle (there are C_3 of these):

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

The first diagonal entry is always 1, the other is 5, 5, 5, 4, 4. So the first three have

$$\mathrm{wt}(T) = \frac{1}{1^{5-1} \cdot 2^{7-5}},$$

and the second two are

$$\mathrm{wt}(T) = \frac{1}{1^{4-1} \cdot 2^{7-4}}.$$

So we get

$$\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8} = 1.$$

If your Young diagram is a $2 \times n$ rectangle, these are in bijections iwth Dyck paths. SO we have some weight on Dyck paths; it's $1/2^k$ where this depends on the number of up-steps in the first two diagonals. You can give a probabilistic interpretation in this case and see how it works for Dyck paths.

Theorem 4.8

$\sum_{\lambda \vdash n} f_\lambda^2 = n!$.

This is not as hard as HLF. There are many nice interpretations of this identity.

For example, in representation theory, this identity has a meaning (which we will discuss later in this class) — in the representation theory of S_n:

Fact — Irreducible representations V_λ of S_n, up to isomorphism, can be labelled by Young diagrams with n boxes, or equivalently $\lambda \vdash n$.

Fact — $f^\lambda = \dim V_\lambda$.

Fact — For every(finite) group, if we sum the squares of dimensions of irreducible representations, then you get the order of the group.

So this identity directly comes from representations of S_n. But as combinatorialists, we like to do things in a more down-to-earth way — we would like to see the identity combinatorially. We would like to find a bijection between $n!$ (permutations of n) and expressions on the left-hand side.

There is a bijection like this, called the *Robinson–Schensted correspondence*. It gives a bijection

$$S_n \to \{(P, Q) \mid P, Q \text{ are SYT of the same shape } \lambda \vdash n\}.$$

This correspondence is important not just because it proves the identity — it has many other implications.

For example, if $w \mapsto (P, Q)$ of shape λ, then λ has many interesting features. The first row of λ has size the maximal increasing sequence, and

This relates to queue-sortable permutations — a permutation can be sorted with m queues if they don't have a decreasing subsequence of length $m+1$. (The subsequence doesn't have to be adjacent.) The length of the first column is the maximal possible size of a *decreasing* subsequence. If $w \to (P, Q)$ then $w^{-1} \to (Q, P)$. And so on — there ar emany nice properties.

One historical remark is that many people call this RSK — Robinson–Schensted–Knuth. But RSK is actually a generalization of this to *semistandard* Young tableau, which we will discuss next time.

We are not going to explain the RS correspondence for the following reason: the original construction makes it very hard to prove these nice properties. So we will want to see a construction that implies all these nice features of RSK for free. So we will present it in a way that makes these nice properties trivial.

§5 September 14, 2022

Last class, we saw the identity

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

One way to prove this is the Robinson–Schensted correspondence, which gives a bijection between $w \in S_n$ and two SYT (P, Q) of the same shape $\lambda \vdash n$.

§5.1 The Robinson–Schensted Correspondence

First we'll see the classical construction.

§5.1.1 Schensted Insertion Algorithm

We'll read the entries of w in order and insert them in order into a tableaux P, called the *insertion tableau*. We'll use Q as the *recording tableau*, which keeps track of the order in which boxes were added.

In the middle of the algorithm, suppose we have an intermediate tableaux T, and we want to insert a. We'll describe this by pseudo-programming language:

1. Set $i := 1$, and $x := a$.
2. If all entries in the ith row are at most x (or if the ith row is empty), then we add a new box at the end of the ith row, and fill it with x, and stop.

So in the first step, we look at the first row. If everything in the first row is at most x, then we add x at the end of the row.

Note that here we have $\leq$, but all entries in a standard Young tableau are different. The reason is that the same algorithm also works in the semistandard case, as we'll see later.

3. Otherwise, find the leftmost entry y in the ith row such that $y > x$. Replace y with x, and set $x := y$ and $i := i + 1$, and go to step 2.

Example 5.1

Suppose our intermediate tableau is

$$T = \begin{array}{|c|c|c|c|}\hline 1 & 4 & 5 & 9 \\ \hline 2 & 7 \\ \cline{1-2} 6 \\ \cline{1-1} 10 \\ \cline{1-1}\end{array}$$

and we want to insert $a = 3$. First we try to insert a in the first row of t. There are some entries strictly greater than 3, and the leftmost is 4. So 3 bumps 4 out of its place, and now we try to insert 4 into the second row.

Now there are still entries greater than 4, the leftmost is 7. So 4 replaces 7, and 7 goes to the third row. Now everything in the third row is at most 7, so we add a new box there. So we get

$$\begin{array}{|c|c|c|c|}\hline 1 & 3 & 5 & 9 \\ \hline 2 & 4 \\ \cline{1-2} 6 & 7 \\ \cline{1-2} 10 \\ \cline{1-1}\end{array}.$$

Theorem 5.2 (Robinson–Schensted Correspondence)

Suppose we have a permutation $w = w_1 \cdots w_n$. Then we start with $P = \varnothing$, and then we insert w_1, w_2, $\ldots$: so

$$P = ((\varnothing \leftarrow w_1) \leftarrow w_2) \cdots \leftarrow w_n.$$

Meanwhile, for Q, when we insert w_i into P and add one new box, we simultaneously add a new box to Q in the same position, filled with i.

So we build Q in the order 1, 2, 3, 4, $\ldots$ (while we build P in the order w_1, w_2, $\ldots$), keeping them the same shape.

Example 5.3

Take the permutation
$$w = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 1 & 6 & 5 & 2 \end{bmatrix} \begin{matrix} \rightsquigarrow P \\ \rightsquigarrow Q \end{matrix}$$
in 2-line notation.

We start with $P = Q = \varnothing$.

Then we insert 3: so we get
$$P = \begin{array}{|c|}\hline 3 \\ \hline\end{array}, Q = \begin{array}{|c|}\hline 1 \\ \hline\end{array}.$$

Then $4 > 3$, so we add to the first row to get
$$P = \begin{array}{|c|c|}\hline 3 & 4 \\ \hline\end{array}, Q = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline\end{array}.$$

Now we want to insert 1 into P; this bumps 3 (the smallest entry strictly greater than 1) from the first row, so now
$$P = \begin{array}{|c|c|}\hline 1 & 4 \\ \hline 3 \\ \cline{1-1}\end{array}, Q = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 \\ \cline{1-1}\end{array}.$$

Then we insert 6 to get
$$P = \begin{array}{|c|c|c|}\hline 1 & 4 & 6 \\ \hline 3 \\ \cline{1-1}\end{array}, Q = \begin{array}{|c|c|c|}\hline 1 & 2 & 4 \\ \hline 3 \\ \cline{1-1}\end{array}.$$

Then we add 5, which bumps 6, so we get
$$P = \begin{array}{|c|c|c|}\hline 1 & 4 & 5 \\ \hline 3 & 6 \\ \cline{1-2}\end{array}, Q = \begin{array}{|c|c|c|}\hline 1 & 2 & 4 \\ \hline 3 & 5 \\ \cline{1-2}\end{array}.$$

Then we insert 2 which bumps 4, so we get
$$P = \begin{array}{|c|c|c|}\hline 1 & 2 & 5 \\ \hline 3 & 4 \\ \cline{1-2} 6 \\ \cline{1-1}\end{array}, Q = \begin{array}{|c|c|c|}\hline 1 & 2 & 4 \\ \hline 3 & 5 \\ \cline{1-2} 6 \\ \cline{1-1}\end{array}.$$

This is reversible — given P and Q, we can recover our permutation. To do so, we first find the maximal entry in Q — this tells us which box was added last. So we know that on the last step, we tried adding 6.

Now we know 6 was bumped by something from the previous row. So we want to find the maximal entry in that row that's strictly smaller than 6.

You can check that all steps are reversible, so this is a bijection.

But this is actually much more powerful, because other things are true.

The first row is the size of the longest increasing subsequence of our permutation. Here the first row had 3 boxes, so we should be able to find exactly three entries in increasing order — here that's 3, 4, 6 for example, or 3, 4, 5. (This doesn't have to be consecutive.) Meanwhile, the first column corresponds to the longest *decreasing* subsequence — here take 6, 5, 2.

Schensted's paper is about increasing and decreasing subsequences; this algorithm helps to solve that problem.

Another feature is if we take the inverse permutation w^{-1}, then the result is (Q, P).

From this construction, the properties look mysterious. In our construction, P and Q play very different roles — it's not symmetric. So it's possible to prove these properties (and they are proved in classical papers — this algorithm is also in the three textbooks).

In general, the first row is the maximal size of the increasing subsequence, the sum of the first two rows is the maximal number of entries that can be covered by *two* increasing subsequences, and so on — the sum of the first k rows is the maximal number of entries that can be covered with k increasing subsequences. The same is true for decreasing subsequences and columns.

It would be nice to find a construction that's *symmetric* — so that the symmetry between P and Q is clear from the construction, since it's kind of mysterious here.

§5.2 Robinson–Schensted–Knuth Correspondence

First we'll see how to generalize to *semi-standard* Young tableaux.

Definition 5.4. A **semi-standard Young tableau** (SSYT) is the same as a Young tableaux, except we allow repeated entries.

Example 5.5

One SSYT is

1	1	1	2	2	4
2	2	4	5	7	
4	5	5	7		

.

Entries can be repeated, and entries can be missing; but entries must be *weakly increasing in rows* and *strictly increasing in columns.*

Remark 5.6. Entries can be arbitrarily large — you could replace 7 with 70.

The **shape** in this example is $\lambda = (6, 5, 4)$. Meanwhile, its **weight** is the sequence $\beta = (\beta_1, \dots)$ where β_i is the number of i's in the tableau — here $\beta = (3, 4, 0, 3, 3, 0, 2)$.

Remark 5.7. Some people use the word "content" instead of "weight". Prof. Postnikov doesn't like that because the word content is reserved for something else — the content of a box is $i - j$, which will be important later.

Fix a number n. This will not be the size of our Young diagrams; it will be the maximal possible entry that we're allowed to use. (In this example, n can be 7, or 8, or so on.)

Now RSK is a bijection between

$$\{n \times n \text{ matrices with entries in } \mathbb{Z}_{\geq 0}\} \leftrightarrow \{(P, Q) \text{ SSYTs of the same shape}\}.$$

We will also fix two vectors $\alpha = (\alpha_1, \dots, \alpha_n)$ and $\beta = (\beta_1, \dots, \beta_n)$ such that $\sum \alpha_i = \sum \beta_i$. Then our $n \times n$ matrices should have column sums $\alpha_1, \dots, \alpha_n$ and row sums $\beta_1, \dots, \beta_n$; and our SSYT should have that the weight of P is α and the weight of Q is β.

This is almost the same as the Schensted correspondence, with a little twist; we'll see an example.

Remark 5.8. Sometimes we have $m \times n$ matrices instead. But it is easier to transform into square matrices — if we have two vectors, we can always put some number of zeroes in the end, which doesn't change anything.

Example 5.9

Take $n = 3$. Suppose we have

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix}.$$

First, we replace the numbers by dots:

$$A = \begin{bmatrix} & \cdot & \\ \cdot\cdot & \cdot & \\ \cdot\cdot\cdot & & \cdot \end{bmatrix}.$$

Then whenever we have a dot in position (i, j), we replace it by the pair $(i, j)^t$. For example, our dots here are $(1, 2)$, $(2, 1)$, $(2, 1)$, $\ldots$. Now arrange these pairs in the lexicographical ordering — we're reading dots by rows left to right. So we get

$$\begin{matrix} 1 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 2 & 1 & 1 & 2 & 1 & 1 & 1 & 3 \end{matrix}.$$

We can think of this as a genearlized permutation.

Now we perform the same algorithm — we insert numbers from the second row into the insertion tableau, and numbers from the first into the recording tableau.

So we have $A \rightsquigarrow \begin{bmatrix} u_1 & u_2 & \cdots & u_N \\ w_1 & w_2 & \cdots & w_N \end{bmatrix}$. Then P is given by the same formula

$$((\varnothing \leftarrow w_1) \leftarrow w_2) \cdots \leftarrow w_N,$$

and when we insert into Q, we add a box filled with u_i. We have to be more careful about weak vs strict inequalities, here.

Example 5.10

For this matrix, to get P we should insert 2, 1, 1, 2, 1, 1, 1, 3. We start with $P = Q = \varnothing$. Then we get

$$P = \begin{ytableau} 2 \end{ytableau}, Q = \begin{ytableau} 1 \end{ytableau}.$$

Then we insert 1 which bumps 2, so

$$P = \begin{ytableau} 1 \\ 2 \end{ytableau}, Q = \begin{ytableau} 1 \\ 2 \end{ytableau}.$$

Then we insert 1 again, so we get

$$P = \begin{ytableau} 1 & 1 \\ 2 \end{ytableau}.$$

Then we insert 2 to get

$$P = \begin{ytableau} 1 & 1 & 2 \\ 2 \end{ytableau}, Q = \begin{ytableau} 1 & 2 & 2 \\ 2 \end{ytableau}.$$

Then we insert 1 again. This bumps the first number that's *strictly* greater, which is 2, so

$$P = \begin{ytableau} 1 & 1 & 1 \\ 2 & 2 \end{ytableau}.$$

Then we add two 1's to get

$$P = \begin{ytableau} 1 & 1 & 1 & 1 & 1 \\ 2 & 2 \end{ytableau}.$$

Then we add 3 as

$$P = \begin{ytableau} 1 & 1 & 1 & 1 & 1 & 3 \\ 2 & 2 \end{ytableau}, Q = \begin{ytableau} 1 & 2 & 2 & 3 & 3 & 3 \\ 2 & 3 \end{ytableau}.$$

Now the size of the first row is the size of the longest weakly increasing subsequence, and the first column is the size of the longest strictly decreasing subsequence. And if $A \mapsto (P, Q)$, then $A^T \mapsto (Q, P)$. If you specialize to the case where alll row and column sums are 1 (so our matrices are permutation matrices), then P and Q are SYT.

This is also a bijection, but we run into some difficulty if we try to invert — there are *several* maximal entries in the Q tableau, so how do we tell which was added last?

Claim 5.11 — These boxes 3 were added from left to right.

So the box that was added last is the *rightmost* box with the maximal entry.

We'll later see another construction for RSK which Prof. Postnikov likes better, and the first step is to transform SSYT into Gelfand–Tsetlin patterns. We have zero minutes to explain what those are, so we will see them next lecture.

Remark 5.12. The shape of P and Q is a partition fo the sum of entries, denoted N.

§6 September 16, 2022

Last time, we discussed RSK — a bijection between nonnegative integer $m \times n$ matrices with fixed column sums α_i and row sums β_j, and SSYT of the same shape λ where the weight of P is α and the weight of Q is β. The statement is for rectangular matrices, but we can assume they are squares by adding columns with sum 0. Today we will assume $m = n$.

Today we will see alternative ways to look at RSK. To do that, we'll define another combinatorial object.

§6.1 Gelfand–Tsetlin Patterns

Definition 6.1. Fix n. A Gelfand–Tsetlin pattern is an array of numbers where the first row is a partition $(\lambda_1, \ldots, \lambda_n)$ (some entries may be 0), the second is a partition $(\mu_1, \ldots, \mu_n)$ such that the entries are interlaced — $\lambda_1 \geq \mu_1 \geq \lambda_2 \geq \mu_2 \geq \cdots$, the second row is the same with ν, and so on up to one number ω_1.

So it's just a triangle of numbers where numbers weakly decrease going southeast and northeast.

We'll abbreviate them by GT patterns.

Proposition 6.2

There is a bijection

$$\{\text{SSYTs with entries in } \{1, \ldots, n\}\} \leftrightarrow \{\text{GT patterns of size } n\}.$$

Note that we do not need to use all entries 1 to n.

We'll do this by example.

Example 6.3

Take the SSYT

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

and suppose $n = 7$.

First, we write the number of 1's in the bottom.

$$3$$

Then we look at the shape formed by the 1's and 2's, and we write that in the next row.

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

$$\begin{array}{ccc}
1 & & 2 \\
 & 3 &
\end{array}$$

Now we look at 1, 2, and 3. We write this as 5, 2, 0 — in general, a SSYT with only 1, 2, 3 has *at most* three rows.

$$\begin{matrix} 6 & 5 & 4 & 0 & 0 & 0 & 0 \\ 6 & 4 & 3 & 0 & 0 & 0 & \\ 6 & 4 & 3 & 0 & 0 & & \\ 6 & 3 & 1 & 0 & & & \\ 5 & 2 & 0 & & & & \\ 5 & 2 & & & & & \\ 3 & & & & & & \end{matrix}$$

Note that we have a lot of 0's — that's because generically a SSYT with numbers up to 7 can have up to 7 rows, but htis one only has 3.

In general, let's call our SSYT T and pattern p. Then the jth entry in the ith row of p (from the bottom) is equal to the number of entries at most i in the jth row of T.

Example 6.4

For the 3 in the 6310 row, that's the 5th row from the bottom, and the third entry. So we should look at the third row, and count entries less than or equal to 5:

1	1	1	2	2	4
2	2	4	5	7	
4	5	5	7		

Question 6.5. Given a GT pattern, how do we find the shape and weight of T?

The shape i seasy — it's just the top row of p (since that captures all of our shape λ).

We know β_1 is the number of 1's, which is exactly the bottom entry. Then $\beta_1 + \beta_2$ is exactly the number of 1's and 2's, which is exactly the second row sum. So in general, $\beta_1 + \cdots + \beta_i$ is the ith row sum of p (from the bottom).

So now we have an alternative way of thinking about SSYT.

Both are arrays of numbers satisfying some inequalities. In SSYT, we had weak inequalities in rows, and strict inequalities in columns. Meanwhile in GT patterns, we only have weak inequalities. This is nicer because we can think of the entries as a collection of $\binom{n}{2}$ variables satisfying some inequalities; if we let the entries be *real*, then we get a polytope. So SSYT correspond to the lattice points on these polytopes. (The SSYT also immediately corresponds to a polytope, but you'd have to remove some faces, which is more complicated — the polyhedral point of view works much better for GT patterns.)

§6.2 RSK for GT Patterns

As we've seen earlier, RSK sends

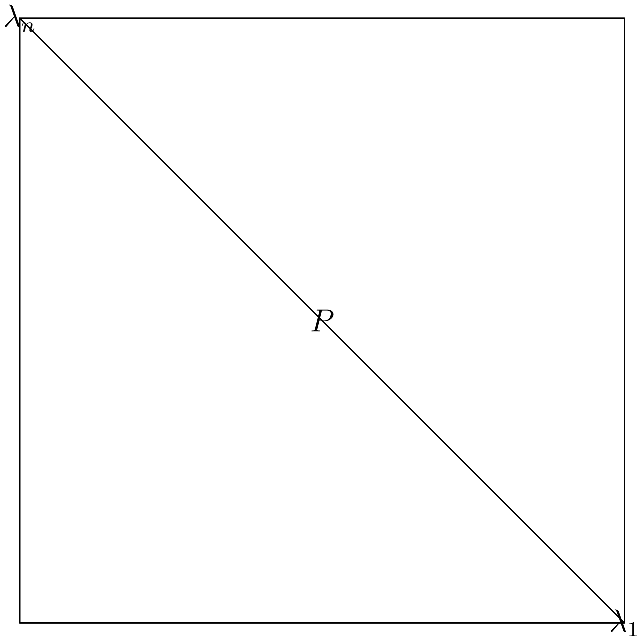
$$A \mapsto (P, Q) \text{ SSYT of same shape } \lambda.$$

Let's assume A is a $n \times n$ matrix. Now we can transform

$$(P, Q) \rightsquigarrow (p, q) \text{ GT-patterns.}$$

Then p and q are both triangular arrays with the weak inequalities, and with the same top row.

If you have two right triangles with the same hypotenuse, we can try to make a square! So we'll glue these two triangles into one square (first we rotate):



Then RSK becomes a map

$$A : \{n \times n \text{ matrices with nonnegative integer entries} \to n \times n \text{squares}$$

with entries weakly increasing in rows and columns (and still with nonnegative integer entries). We call these **reverse plane partitions** (plane partitions are the same thing but with numbers weakly decreasing — they're a generalization of one-dimensional partitions).

Call the column sums $\alpha_1, \ldots, \alpha_n$, and the row sums $\beta_1, \ldots, \beta_n$.

Recall that the weight entries correspond to row sums in the GT pattern, which become the diagonal sums. SO the weight of P is given by differences of the diagonal sums.

Suppose the diagonal sums of the RPP are $d_1, d_2, \ldots, d_{2n-1}$. Then:

- $d_1 = \alpha_1$;
- $d_2 = \alpha_1 + \alpha_2$;
- $d_3 = \alpha_1 + \alpha_2 + \alpha_3$;
- $\cdots$
- $d_n = \alpha_1 + \alpha_2 + \alpha_3 + \cdots + \alpha_n = \beta_1 + \cdots + \beta_n$;
- $d_{n+1} = \beta_1 + \cdots + \beta_{n-1}$;
- $d_{n+2} = \beta_1 + \cdots + \beta_{n-2}$;
- $\cdots$
- $d_{2n-1} = \beta_1$.

Let's call this map ϕ_n^{RSK}. Remember that we already constructed a map like this.

Claim 6.6 — The construction from the previous lecture is exactly equivalent to RSK! In other words, $\varphi_n^{\text{RSK}} = \varphi_{n\times n}$.

Note that $\varphi_{n\times n}$ is a square partition. So λ from the previous lecture plays quite a different role.

Let's rephrase the toggle construction of this map in a slightly different language. We'll now assume our array is just a square, to keep things simple (this works for an arbitrary shape).

We'll call this construction *(generalized) Fomin's growth diagrams* — Sergey Fomin gave an interpretation of the Robinson–Schensted correspondence using growth diagrams, and this is a generalization to semistandard Young tableaux.

$$
\begin{array}{|c|c|c|c|c|}
\hline
a_{11} & & & & a_{1n} \\
\hline
 & & & & \\
\hline
a_{n1} & & & & a_{nn}
\end{array}
$$

These things have boxes, and they also have nodes (the intersections between lines). In the boxes, we just put our numbers.

Then we assign a *partition* to every node in this diagram, according to a rule. This shoul dhave the feature that if we have two horizontally or vertically adjacent nodes, with λ and μ, then the parts of λ should be interlaced with the parts of μ (same as in GT) — we should have $\mu_1 \geq \lambda_1 \geq \mu_2 \geq \lambda_2 \geq \cdots$ — the size of partitions increases going right and down.

Here are our rules:

- For every node on the left and top border, we have the empty partition.

  Now we start filling the other ones one-by-one (this is why they're called growth diagrams — we start with empty partitions and grow t his thing until we fill in all nodes):

Now here is our rule: For each box of the diagram, let's suppose it has a. THen it has four nodes λ, μ, ν, and (λ NW, μ SW, ν NE). Now we want to use these three to find the other partition λ^{toggle}. This is basically given by the toggle rule — write
$$
\begin{array}{ccccc}
\mu_1 & & \mu_2 & & \\
 & \lambda_1 & & \lambda_2 & \\
\nu_1 & & \nu_2 & &
\end{array}
$$

Now we toggle λ in the exact same way as before: we produce λ^{toggle} as

$$
\begin{array}{ccccc}
 & \mu_1 & & \mu_2 & \\
\lambda_0^* & & \lambda_0^* & & \\
 & \nu_1 & & \nu_2 &
\end{array}
\qquad \text{(flip all the signs) where}
$$

$$\lambda_0^* = \max(\mu_1, \nu_1) + a$$

and all others are toggles —
$$\lambda_i^* = \min(\mu_i, \nu_i) + \max(\mu_{i+1}, \nu_{i+1}) - \lambda_i.$$

Using this, we fill the whole growth diagram with partitions.

Now we want to read P and Q from this growth diagram.

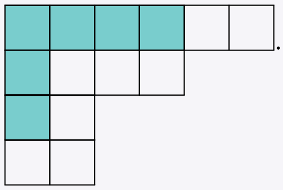
$$(a_{ij}) \rightsquigarrow \text{toggles} a_{ij} \text{ with nodes.}$$

Now if we look at the bottom side, we get an increasing sequence of partitions. This is actually our P-tableau — this row of increasing partitions is the P-tableau from RSK, where the ith entry is the tableau formed by all things at most i in P. The same is true for the right side and the Q-tableau.

Student Question. Can you also interpret the other rows/columns?

Yes, they're intermediate shapes in RSK. We'll see this next time.

Definition 6.7. Suppose we have a SYT, and we look at all shapes formed by all entries up to $i-1$.



Then we can look at the i's. These form a **horizontal strip** — every column contains at most one box.

So we can think of SSYTs as ways to grow our Young tableaux by adding horizontal strips (that's actually how you construct GT patterns). How to tell whether going $\lambda \to \mu$ by adding a horizontal strip? That's exactly the interlacing condition!

Note that Sergey Fomin's original paper (about the standard case, where A is a permutation matrix and all tableaux in the end are SYT) involved several cases — there were 5 different ways of filling the square, depending on whether two of the partitions are equal or obtained by various ways of adding boxes. But somehow, once we formulate it in terms of toggles, all these cases correspond to the same toggle move — so the generlaization from this point of view i sactually simpler (the toggle point of view unifies the cases).

Exercise 6.8. If we use Gelfand–Tsetliln patternsn, you can identify SSYT with lattice points in polytopes, called Gelfand–Tsetlin polytopes.

There are several versions. One is to look at all GT patterns where we allow arbitrary real entries instead of integers, but we fix the top row $\lambda_1, \ldots, \lambda_n$ (the rest are our variables). Call this GT_λ.

Then there is another version where you fix λ and you also fix the weight (so we fix the row sums). Call that $GT_{\lambda,\beta}$. Assuming all entries of λ and β are integers, are these alway sinteger polytopes (meaning their vertices are integers)?

For example, suppose all parts of λ are integers, and ifnd all vertices of these polytopes. We want to find all vertices.

A vertex means we have a bunch of inequalities, and we want to make as many of these into equalities. If you transform one that's a facet, and so on; if you impose as many equalities as possible (you can't make all of them unless you have trivial patterns, but you can make as many as possible) and these will be the vertices. YOu want to combinatorially describe these vertices.

§7 September 19, 2022

We can think of Schensted insertion as the building block for RSK. Here we've broken every insertion step into smaller steps — if Schensted insertion is molecules, then toggles are atoms. It's left as an exercise to check that these two constructions are equivalent — Schensted insertion decomposes into a sequence of toggles.

But now there's a clear symmetry between P and Q — if we transpose A (flipping on the main diagonal), we simply swap P and Q. This is hard to show in the classical construction.

Example 7.1

Take the matrix

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix}.$$

Then the nodes in our growth diagram (at the corners) are

$$\begin{array}{cccc} \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & (1) & (1) \\ \varnothing & (2) & (3,1) & (3,1) \\ \varnothing & (5) & (5,2) & (6,2,0) \end{array}$$

For example, to produce the second row, last thing: we write

$$\begin{array}{ccccccc} 0 & & 0 & & 0 & & 0 \\ & 0 & & 0 & & 0 & \\ 1 & & 0 & & 0 & & \end{array}$$

Then we put $\max(0,1) + 0$ on the left, and all the toggles produce 0, so we get (1).

For the first nontrivial thing, we have

2 0 0 0 0 0 1 0 0 So we toggle our last zero in the interval $[0,1]$ to get 1; and we take $\max(2,1) + 1 = 3$.

We write $(6,2,0)$ because a priori it might have three parts.

Now how to read off P and Q? Looking at the bottom row we write down the GT pattern

$$6,2,0;5,2;5,$$

which corresponds to

$$P = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 1 & 3 \\ \hline 2 & 2 \\ \cline{1-2} \end{array}.$$

Similarly the right thing represents Q. So the GT pattern is

$$620;31;1$$

which corresponds to

$$\begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 2 & 2 & 3 & 3 & 3 & 3 \\ \hline 1 & 3 \\ \cline{1-2} \end{array}.$$

This is exactly the same as RSK; the general proof is left as an exercise. You should check that each Schensted insertion can be obtained by a composition of toggles.

§7.1 Greene's Theorem

Question 7.2. What is the shape of P and Q?

The classical formulation of Greene's Theorem is: Suppose $A = (a_{ij})$. Then RSK works by transforming A into a generalized permutation — we take the multiset of pairs (i, j) repeated a_{ij} times, so here we have one $(1, 2)$, two $(2, 1)$, and so on; then we arrange them lexicographically as

$$\begin{matrix} 1 & 2 & 2 & 2 & \cdots \\ 2 & 1 & 1 & 2 & \cdots \end{matrix}$$

to get a "generalized perutation"

$$\begin{bmatrix} i_1 & i_2 & \cdots & i_N \\ j_1 & j_2 & \cdots & j_n \end{bmatrix}.$$

with $i_1 \leq i_2 \leq \cdots$ (and if $i_k = i_{k+1}$ then $j_k \leq j_{k+1}$).

Theorem 7.3 (Greene's Theorem)

λ_1 is the size of the maximal weakly increasing subsequence in $j_1, \ldots, j_N$.

$\lambda_1 + \lambda_2$ is the maximal cardinality of the subset of all indices in $\{1, \ldots, N\}$ that can be covered by *two* weakly increasing subsequences.

$\lambda_1 + \lambda_2 + \lambda_3$ is the same with 3, and so on — $\lambda_1 + \cdots + \lambda_k$ is the maximal cardinality that can be covered by k increasing subsequences.

There is something nontrivial here — when you have the largest increasing sequence, then you can try to fix it and find another increasing subsequence in the remaining set of indices, and hope that's λ_2. But this is not always true — once you fix one particular maximal increasing subsequence, the maximal increasing subsetquence in the remaining may be smaller than λ_2.

There is something about this formulation that is not very satisfactory — we have broken symmetry between the rows and columns. So it would be nicer to formulate this rule in a more intrinsic way, without converting A to a permutation.

We know λ is exactly the partition we put in the final corner. But we could also ask more generally, if we can find a non-recursive rule for *any* partition in the growth diagram. Suppose we have a node in position (k, ℓ) — what partition should we put here?

The way the growth diagrams are constructed, this only depends on the rectangular submatrix from the start to (k, ℓ). In fact, the rule for this partition μ in terms of the submatrix is the same for the rule for λ for the whole mtarix. So it is equally easy to find any intermediate matrix as the original.

A maximal weakly increasing subsequence is: consider our $k \times \ell$ submatrix $A^{k,\ell}$ of A. Then we need to find a lattice path from a_{11} to $a_{k\ell}$; there are a binomial coefficient number of these.

Now μ_1 is the maximum, over all lattice paths from $(1, 1)$ to (k, ℓ), of $\sum_{i,j \in \mathcal{P}} a_{ij}$. Then $\mu_1 + \mu_2$ is the maximum sum in a pair of *non-crossing* lattice paths. And so on:

Theorem 7.4

$\mu = (\mu_1, \ldots, \mu_r)$ where $r = \min(k, \ell)$, and $\mu_1 + \mu_2 + \cdots + \mu_s$ is the maximum over all subsets $S \subseteq [k] \times [\ell]$ that can be covered by s non-crossing lattice paths connecting the points $(1, 1)$, $(2, 1)$, $\ldots$, $(s, 1)$ with $(k, \ell - s + 1)$, $\ldots$, (k, ℓ), of the sum of entries:

$$\mu_1 + \cdots + \mu_s = \max_{S \subseteq [k] \times [\ell] \text{ covered by } s \text{ non-crossing LP}} \sum_{(i,j) \in S} a_{ij}.$$

Question 7.5. Why do we need them to be non-crossing and to start from different points?

We don't, but we don't want to doublecount; if they're crossing you can remove points from them.

By non-crossing, we mean on the dots — where the dots are a_{ij}.

From a tropical geometry point of view, this is the right thing to think about.

> **Exercise 7.6.** Prove this — show that this non-recursive rule for the entry in teh growth diagarm is equivalent to the toggle rule.

THis shouldn't be terribly hard — we have the a_{ij} and then we construct a bunch of partitions given by this rule, so the only thing we need to show is that the array of partitions satisfies the local toggle rule. This is a local verification you can do by hand; it requires a bit of work but is not terribly hard.

The "correct" way to thinik about this si in terms of tropical geometry.

§7.2 Tropicalization

In rational algebra, we have three operations: $\cdot$, $/$, and $+$. (We assume we do not use $-$.) But here we have different operations — we add and we take maximums.

But there is something called *tropical semiring* that also has three operations:

$$\begin{array}{c|c} \cdot & + \\ / & - \\ + & \max \end{array}$$

If you have something of the first type, you can make all these replacements, and **tropicalize** it — tropicalization is a way to go from rational algebra to teh piecewise linear world. (You do not need to take minima, becaues $\min(a, b) = -\max(-a, -b)$.)

> **Student Question.** Why the name tropical?
>
> It was originally discovered by Brazilian mathematics, which is close to the tropics.

So there is a tropicalization operation. Meanwhile you can go back via **geometrization** or **detropicalization**. This is a bit less straightforward because it's not necessarily well-defined — there are many ways to write down the same pieceiwse linear thing, so you might get different results. So you need to make sure your expression is written in the correct form.

From this point of view, our expression is the correct form. There's another way to write down our thing — take lattice paths and all subsets of lattice paths. (Now we're just adding smaller terms, so we should get the same subtraction free expression. But when you detropicalize you get a messy thing.)

Now if you take the expression given by this thing and detropicalize it, you get the sum over s-tuples of lattice paths, of some products.

There is a famous ? lemma we will discuss later, that once you have an expression sum over families of non-crossing lattice paths, this equals the determinant of a certain matrix. So when you detropicalize, you get a bunch of minors of a certain matrix. The toggle relation becomes a certain rational relation between minors, and that's exactly the Pluker relation.

Right now we have several combinatorial constructions that may look unmotivated. But all these stuff show up in many applications in algebra and geometry, as we'll see later.

§7.3 Schur Polynomials

The **Schur polynomials** are polynomials $s_\lambda(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]^{S_n}$ (symmetric polynomials). They will form a basis for that, and they play many important roles.

Sometimes it is convenient to fix the number of variables; sometimes we look at $s_\lambda(x_1, x_2, \ldots)$ with infinitely many variables, which are called **Schur functions** instead of Schur polynomials. This is because by definition, a polynomial contains finitely many terms, but these Schur functions will have infinitely many terms and not be polynomials. We can easily go between these two things by fixing n variables and setting the rest to 0.

We will see four different definitions of Schur polynomials.

1. The combinatorial definition —

$$s_\lambda(x_1, \ldots, x_n) = \sum_{T \text{ SSYT with entries in } [n]} x^{\text{weight}(T)}.$$

By this we mean x_1 to the number of 1's, x_2 to the number of 2's, and so on.

> **Fact 7.7** — This is symmetric — if we permute the entries of β, the number of SSYT remains teh same.

2. The *classical definition* (how these were originally defined) — the determinant of some matrix divided by the determinant of another matrix. In this definition, from representation theory this is called the **Weyl character formula**, and is closely related to representation theory of GL_n.
3. Another definition from geometry — s_λ is a certain Schubert polynomial (we will discuss these things later, they're a more general class of Schur polynomials, and in particular usual Schur polynomials are special cases of Schubert, and described by divided difference operators).
4. The mazur character formula.

§8 September 21, 2022

§8.1 Schur Polynomials and Functions

Schur polynomials are the polynomials $s_\lambda(x_1, \ldots, x_n) \in \Lambda_n = \mathbb{C}[x_1, \ldots, x_n]^{S_n}$, where λ is a partition with at most n parts. The Schur polynomials form a basis for this ring of symmetric polynomials.

The **Schur functions** $s_\lambda(x_1, x_2, \ldots)$ have infinitely many variables. These form a basis for $\Lambda := \lim_{n\to\infty} \Lambda_n$ (we won't define what this "limit" means; it has to do with graded rings). More explicitly, Λ is the set of formal series in infinitely many variables $x_1, x_2, \ldots$ of bounded degree, which are invariant under permutation of the variables x_i.

Example 8.1

$x_1 + x_2 + \cdots \in \Lambda$. But $x_1 x_2 x_3 \cdots$ is not, since it does not have bounded degree.

We will today mostly talk about symmetric polynomials in n variables.

Definition 8.2 (Combinatorial Definition).

$$s_\lambda(x_1, \ldots, x_n) = \sum_{T \text{ SSYT of shape } \lambda \text{ filled with entries in } [n]} x^{\text{weight}(T)}.$$

We will write s_λ^{comb} for this definition. Here $x^{\text{weight}(T)}$ means $\prod_{i=1}^n x_i^{\#\text{ of } i}$.

Example 8.3

We have $s_{\square}(x_1, \dots, x_n) = x_1 + x_2 + \cdots + x_n$, since we can take a single box and place any number inside it.

From this definition and the fact that we have already proved RSK, we immediately get an important formula for Schur polynomials, known as the *Cauchy formula*:

Proposition 8.4 (Cauchy Formula)

We have

$$\sum_{\lambda} s_\lambda(x_1, \dots, x_m) \cdot s_\lambda(y_1, \dots, y_n) = \prod_{i=1}^{m} \prod_{j=1}^{n} \frac{1}{1 - x_i y_j}.$$

We need to make sure both s_λ make sense — so λ has to be a partition with at most $\min(m, n)$ parts. (If it has more, one of the s_λ is automatically 0.)

Proof. If you expand both sides, you get combinatorial objects which are in bijection by RSK. Expanding the LHS, we get

$$\sum_{(P,Q) \text{ SSYT of shape } \lambda} x^{\text{weight}(P)} y^{\text{weight}(Q)}.$$

On the RHS we expand using geometric series — we then get

$$\prod_{i,j} \sum_{a_{ij} \geq 0} (x_i y_j)^{a_{ij}}.$$

So you get arrays of these a_{ij}, which correspond to matrices — we can write this as

$$\sum_{A\ m \times n \text{ matrices with nonneg int entries}} x^{\text{column sums}} y^{\text{row sums}}.$$

This is exactly the objects we have in RSK. $\square$

In fact, this formula almost defines Schur polynomials — they're the unique collection of polynomials satisfying this formula plus some extra simple conditions. So you can *define* Schur polynomials almost based on this identity.

Definition 8.5 (Classical Definition)**.** Let $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$, and for this nonnegative integer vector, define the generalized Vandermode determinant

$$a_\alpha = \det \begin{bmatrix} x_1^{\alpha_1} & \cdots & x_n^{\alpha_1} \\ x_1^{\alpha_2} & \cdots & x_n^{\alpha_2} \\ \vdots & \vdots & \vdots \end{bmatrix}.$$

Notice that tis determinant is not symmetric — it is antisymmetric (if you switch two variables, the sign changes). It isi also antisymmetric with respect to permutations of α — so it is doubly antisymmetric.

Because it is antisymmetric with respect to permutations of α, if two entries of α are equal then it's zero. So we can WLOG assume that all entries of α are distinct, and are arranged in strictly decreasing order — $\alpha_1 > \alpha_2 > \cdots > \alpha_n$. Now let

$$\delta = (n-1, n-2, \dots, 1, 0)$$

(in Lie theory people usually use ρ). Then we can assume $\alpha = \lambda + \delta$, where δ is this and λ is a regular partition ($\lambda_1 \geq \lambda_2 \geq \cdots$).

It is also clear that a_δ is the determinant of the usual Vandermonde matrix, and

$$a_\delta = \prod_{i<j} (x_i - x_j).$$

Now the **classical definition** is

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

We'll denote these by s_λ^{class}.

It's easy to see that $\alpha_\delta \mid a_{\lambda+\delta}$ — by the anti-symmetry, $a_{\lambda+\delta}$ has to be divisible by all pairs $x_i - x_j$. Moreover, since both are antisymmetric, the result is symmetric. So this is really a symmetric polynomial.

Example 8.6

If $n = 2$, and $\lambda = (1, 0) = \square$, then $\lambda + \delta = (2, 0)$, and so

$$s_\lambda^{\text{class}}(x_1, x_2) = \frac{\begin{bmatrix} x_1^2 & x_2^2 \\ 1 & 1 \end{bmatrix}}{\begin{bmatrix} x_1 & x_2 \\ 1 & 1 \end{bmatrix}} = \frac{x_1^2 - x_2^2}{x_1 - x_2} = x_1 + x_2.$$

We get the same thing, that's not a coincidence.

Theorem 8.7

$s_\lambda^{\text{comb}} = s_\lambda^{\text{class}}$.

We will skip the proof for now and return to it later. First we will see several other formulas.

§8.2 The Symmetric Group

Fact 8.8 — S_n is generated by s_1, s_2, $\ldots$, s_{n-1}, the **adjacent transpositions** — s_i transposes i and $i+1$.

In the more general theory of Coxeter groups, these are called *simple reflections*.

Fact 8.9 — These generators satisfy three relations:

1. $s_i^2 = \mathrm{id}$
2. $s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}$.
3. $s_i s_j = s_j s_i$ for $|i-j| \geq 2$. Two adjacent transpositions with non-adjacent indices commute.

The most interesting is relation 2.

Of course, this means we can write any permutation as a product of simple transpositions, as

$$w = s_{i_1} s_{i_2} \cdots s_{i_\ell}.$$

We can do this in many different ways, and all such ways are related to each other by a sequence of these relations. For example, we can replace $s_1 s_2 s_1$ by $s_2 s_1 s_2$, and so on.

Definition 8.10. $w = s_{i_1} \cdots s_{i_\ell}$ is a **reduced decomposition** of w if ℓ is as small as possible.

So if it has the shortest possible length, we call such things reduced decompositions, and ℓ is called the length of w.

Lemma 8.11

$\ell(w)$ is the number of inversions in w.

Definition 8.12. An **inversion** of w is a pair (i, j) with $1 \leq i < j \leq n$ and $w(i) > w(j)$.

The second lemma is fundamental:

Lemma 8.13

Any two reduced decompositions of w can be obtained from each other by a sequence of moves (2) and (3).

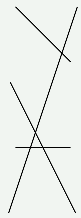
Any two decompositions can be obtained from each other by using (1), (2), and (3); this lemma says that for reduced ones, we never have to use (1). (This isn't obvious because you may need to increase the length and then decrease it again, and this lemma says that you don't need to do that — you can always transform one to another without increasing the length of your transposition.) This is a good exercise to prove combinatorially.

§8.2.1 Wiring Diagrams

We can geometrically describe reduced decompositions by wiring diagrams.

Example 8.14

Take $w = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 2 & 1 & 3 \end{pmatrix}$. We can represent this by the following picture:



If you now scan all possible intersections, we see an intersection at the bottom level, so that's s_2. And so on. We can read $s_1 s_2 s_1 s_3$ in order. So this picture represents $w = s_3 s_1 s_2 s_1$ (we read backwards, because that means we first apply s_1, then s_2, ... in the sense of function composition).

Fact 8.15 — Such a decomposition is reduced if and only if this picture has no double crossings — any pair of wires intersects at most once.

What possible transofrmations of wiring diagrams can we do? We can replace $s_1 s_3$ by $s_3 s_1$. That's because we have two crossings between two different pairs of wires; we can move one of them to the right (topologically you can do that).

You can replace $s_1 s_2 s_1$ by $s_2 s_1 s_2$ — that means you move the 2 wire above the intersection fo teh other two. The first gets rid of a consecutive double crossing. So it's clear that generically only these three things can happen.

Then this lemma says that if you know your diagram has no double crossings, you can modify it to antoher such thing without introducing any double crossings.

§8.3 Divided Difference Operators

These are operators ∂_1, ∂_2, ..., ∂_{n-1} acting on $\mathbb{C}[x_1, \dots, x_n]$ by the following:

$$\partial_i \colon f(x_1, \dots, x_n) \mapsto \frac{1}{x_i - x_{i+1}}(1 - s_i)(f).$$

Here $1 \cdot f$ is just f; while $s_i f$ means s_i acts on f by permuting the variables. So we take $f(x_1, \dots, x_n) - f(x_1, \dots, x_{i+1}, x_i, \dots, x_n)$, where we switched x_i and x_{i+1} (and did nothing else), and after that we divide by $x_i - x_{i+1}$. It's clear the result is a polynomial, because if $x_i = x_{i+1}$ then $f = s_i f$, so the numerator is zero.

Lemma 8.16

These operators satisfy the relations:

(1) $\partial_i^2 = 0$.

(2) $\partial_i \partial_{i+1} \partial_i = \partial_{i+1} \partial_i \partial_{i+1}$.

(3) $\partial_i \partial_j = \partial_j \partial_i$ if $|i - j| \geq 2$.

These are almost the same as the relations for S_n. The only difference is that in our first thing $s_i^2 = \mathrm{id}$, but here $\partial_i^2 = 0$.

But if we stay within the class of reduced decompositions, we never need to use the first relation — we can just use the last two to connect any two reduced decompositions.

And that means we can define, for any $w = s_{i_1} \cdots s_{i_\ell}$ a reduced decomposition, we can define $\partial_w := \partial_{i_1} \cdots \partial_{i_\ell}$. And we know the result will be independent of the choice of the decomposition, since the rules are the same for the ∂_i and the simple reflections. So ∂_w only depends on w, and not the choice of reduced decomposition.

Question 8.17. What happens if you take a decomposition that is not reduced?

Then you get 0 — it's possible to apply some sequence of moves to get to a place where you have $s_i s_i$, and then we get $\partial_i \partial_i = 0$.

Definition 8.18. w_0 is the longest permutation in S_n — $\begin{pmatrix} 1 & 2 & \cdots & n \\ n & n-1 & \cdots & 1 \end{pmatrix}$.

Now we are ready to give another definition. There is a more general class of polynomials, called **Schubert polynomials**, and one way to define them is by using divided difference operators. These are more general than Schur polynomials, so Schur ones are defined by these as well.

Definition 8.19 (Schur as Schubert)**.** $s_\lambda^{\text{schub}} = \partial_{w_0}(x^{\lambda+\delta})$.

Here λ is a partition with n parts, and $\delta = (n-1, n-2, \ldots, 0)$ is the staircase partition.

Theorem 8.20

$s_\lambda^{\text{schub}}(x_1, \ldots, x_n) = s_\lambda^{\text{class}}(x_1, \ldots, x_n)$.

It turns out ∂_{w_0} is very closely related to the classical definition. The formula that will immediately imply our theorem is hte following:

Theorem 8.21

$\partial_{w_0} = \frac{1}{\prod_{i<j}(x_i - x_j)} \cdot \sum_{w \in S_n} (-1)^{\ell(w)} w$.

Here we are working int he group algebra of S_n.

Example 8.22

For $n = 3$, on the LHS we have $\partial_1 \partial_2 \partial_3 = \frac{1}{x_1 - x_2}(1 - s_1) \frac{1}{x_2 - x_3}(1 - s_2) \frac{1}{x_1 - x_2}(1 - s_1)$, where this is an operator acting on polynomials. The theorem says this equals

$$\frac{1}{(x_1 - x_2)(x_2 - x_3)(x_1 - x_3)} \cdot (1 - s_1 - s_2 + s_1 s_2 + s_2 s_1 - s_1 s_2 s_1).$$

You can try to expand the LHS and RHS and see that they are the same thing.

Now we claim that if you apply this formula to a monomial,

$$\partial(x^{\lambda+\delta}) = \frac{1}{\text{Vand}} \sum (-1)^{\ell(w)} \cdot w(x^{\lambda+\delta}).$$

If you thikn about this expression, you see that this is exaclty the generalized Vandermode determinant $a_{\lambda+\delta}$ in the classical definition. So this formula is saying these two things are equivalent.

Of course, ti remains to prove this formula; that will be left as an exercise.

There is one more formula, in terms of Mazur characters, which we will see next time.

§9 September 26, 2022

As we've seen last time, there are many ways to define Schur polynomials. We'll use the combinatorial definition as the base —

$$s_\lambda(x_1, \dots, x_n) = \sum_{T\text{SSYT of shape } \lambda \text{ with all entries } \leq n} x^{\text{weight}(T)}.$$

We can write this similarly as

$$\sum_{\beta=(\beta_1,\dots,\beta_n)} K_{\lambda\beta} x^\beta,$$

where $K_{\lambda\beta}$ is the number of SSYT of shape λ and weight β (here λ is a partition, and β is an arbitrary weight — if we want β to be a partition, then we write it as μ). These are called **Kostke numbers**.

> **Lemma 9.1**
>
> $s_\lambda(x_1, \dots, x_n)$ is a symmetric polynomial in $x_1, \dots, x_n$.

Equivalently, the Kostke numbers $K_{\lambda\beta} = K_{\lambda\tilde{\beta}}$ for any $\tilde{\beta}$ which is a permutation of β.

First we will prove a weaker claim:

> **Lemma 9.2**
>
> $K_{\lambda\beta} = K_{\lambda\tilde{\beta}}$ for $\tilde{\beta}$ obtained by swapping β_{i+1} and β_i, for any $1 \leq i \leq n-1$.

Proof. We need a bijection between SSYT of shape λ with weight β, and SSYT of shape λ with weight $\tilde{\beta}$, for $\tilde{\beta}$ obtained by an adjacent transposition.

We have the entries $1, \dots, i-1$; this gives some shape. Then we have some i's in horizontal strips, and then some $i+1$'s in horizontal strips.

x	x	x	i	i	i i	+ i1	+ 1
x	x	i i	+ 1				

We want to do some transformation of this tableau so that we switch the number of red and blue boxes.

The rule is: first, we don't touch any boxes filled with other entries — we *only* touch boxes filled with i's and $i+1$'s. Also, there are some boxes whose entries we cannot change — we need to ignore all "blocked pairs" of boxes. This means if you have a red box filled with i, and immediately below it is a blue box filled with $i+1$, then you *cannot* change them — because we need a strict inequality.

Meanwhile, for everything else, we do the following: for any row of remaining boxes filled with i's and $i+1$'s, it must have the form $[redboxes(i)][blueboxes(i+1)]$. Say we have a i's and b $(i+1)$'s. So then what we need to do with the seboxes is change this row to have b i's and a $(i+1)$'s.

Clearly then the number of i's and $i+1$'s switch — we're not touching the blocked pairs, and these swap i-counts with $(i+1)$-counts. It's also clear that if you apply this operation twice you get the original tableau, so it's an invertible operation that swaps the number of i's and $(i+1)$'s.

These operations $T \to \tilde{T}$ are called **Bender–Knuth operations**. We call them BK_i.

Example 9.3

Suppose we have a tableau

$$T = \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 2 & 2 & 4 & 4 & 4 & 5 \\ \hline 2 & 2 & 2 & 3 & 3 & 4 & 5 \\ \cline{1-7} 3 & 3 & 5 & 5 & 5 \\ \cline{1-5} 4 & 5 \\ \cline{1-2} 5 \\ \cline{1-1} \end{array}$$

with $\beta = (4,5,4,5,7)$, and suppose $i = 3$, so we are only looking at entries 3 and 4:

$$T = \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 2 & 2 & 4 & 4 & 4 & 5 \\ \hline 2 & 2 & 2 & 3 & 3 & 4 & 5 \\ \cline{1-7} 3 & 3 & 5 & 5 & 5 \\ \cline{1-5} 4 & 5 \\ \cline{1-2} 5 \\ \cline{1-1} \end{array}.$$

Then we ignroe the 34 ont he left because it's blocked. Other than that, we swap the counts. So we get

$$\tilde{T} = \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 2 & 2 & 3 & 3 & 3 & 5 \\ \hline 2 & 2 & 2 & 3 & 4 & 4 & 5 \\ \cline{1-7} 3 & 4 & 5 & 5 & 5 \\ \cline{1-5} 4 & 5 \\ \cline{1-2} 5 \\ \cline{1-1} \end{array}.$$

Its clear $\tilde{\beta} = (4,5,5,4,7)$.

□

We have to be careful — how can we define the action of the symmetric group on the weight? Suppose we want a canonical bijection between SSYT of weight β and *some* permutation (not necessarily adjacent transposition). Here we would have to be a bit careful because the Bender–Knuth operations *do not* satisfy the Coxeter relations. So you can construct a bijection between these two SSYT, but it will not be canonical — it will depend on how you write your permutation as a product of adjacent transpositions. (But it's possible in a smarter way.)

Let's now rewrite this in terms of GT pattersn.

Example 9.4

$T \sim$

$$\begin{array}{ccccc} 10 & 7 & 5 & 2 & 1 \\ 9 & 6 & 2 & 1 & \\ 6 & 5 & 2 & & \\ 6 & 3 & & & \\ 4 & & & & \end{array}$$

and $\widetilde{T} \sim \begin{array}{ccccc} 10 & 7 & 5 & 2 & 1 \\ 9 & 6 & 2 & 1 & \\ 9 & 4 & 1 & & \\ 6 & 3 & & & \\ 4 & & & & \end{array}$

If you look at this, you notice they are very similar — the only difference is in the third row 652 and 941, since that's the only thing you modify. In general, only the ith row from teh bottom will be different. And the rule is exactly the toggle rule! You toggle each entry with respect to its neighbors.

Lemma 9.5

$BK_i(T)$ is obtained by toggling all entries in the ith row (from the bottom) of the corresponding GT pattern.

Recall that in order to toggle x, suppose we have

$$\begin{array}{ccc} a & & b \\ & x & \\ c & & d \end{array}.$$

Then we need x to be in $[\min(a, b), \max(c, d)]$, which is an interval that x should sit in. Then we reflect x with respect to teh middle of this interval. For some boundary entries there's only three neighbors, then you ignore the empty entry. (For example, we're toggling the first 6 in $[6, 9]$.) For example, swapping 5 in the interval $[3, 6]$ becomes 4.

The proof will be left as an exercise.

Remark 9.6. The better way to define the action of S_n was to conjugate BK_i by previous BK operations and if you do it carefully, after that they satisfy Coxeter relations. This was in a paper by Berenstein–Kirillov, around 2000 (in a paper about the action of S_n on SSYT). But for our purposes, just to know Schur polynomials are symmetric, this is enough.

§9.1 Other Definitions of Schur Polynomials

Definition 9.7. The classical construction of $s_\lambda^{\text{class}}(x_1, \ldots, x_n) = \frac{a_{\lambda+\delta}}{a_\delta} = \sum_{w \in S_n} \frac{(-1)^{\ell(w)} w(x^{\lambda+\delta})}{\prod(x_i - x_j)}$, where δ is the staircase partition and w means you permute all x_i's by the permutation w.

In the combinatorial formula we could assume there's infinitely many variables if we wanted to but here we can't.

Definition 9.8 (Schur as Schubert). $S_\lambda^{\text{Schub}}(x_1, \ldots, x_n) = \partial_{w_0}(x^{\lambda+\delta})$.

Recall that ∂_i are the **divided difference operators** sending a polynomial f to $\frac{1}{x_i - x_{i+1}}(1 - s_i)(f)$, so we take f and subtract f with x_i and x_{i+1} swapped. For the longest permutation w_0, this is obtained by taking the product of divided differences for any reduced decomoposition of w_0. This doesn't depend on the choice of decomposition because ∂_i satisfy all the same relations as the Coxeter relations, except that $\partial_i^2 = 0$ instead of the identity.

These relations have a special name (for the ∂_i) — the **nil Coxeter relations** (theyr'e teh same as Coxeter relations but replacing 1 by 0).

There is another formlua for Schur polynomials based on a different idea as well. It's not hard to see that this is equivalent to the classical construction. But we're starting with a monomial and anti-symmetrizing it, and then dividing by Vandermode to make it symmetric.

But we could try to do this in a more simple way — instead of anti-symmetrizing we could try to symmetrize.

Start with a monomial $x^a y^b$. We want to transform it into $x^a y^b$, but we also want $x^b y^a$, and really everything in between — $x^{a-1}y^{b+1} + \cdots + x^b y^a$ (assume $a > b$). This is now symmetric. So we are symmetrizing in a saturated way.

Notice that this is a geometric progression, so if you do a bit of calculations you can rewrite this as

$$\frac{x^a y^b - \frac{y}{x} x^b y^a}{1 - y/x}.$$

This looks a bit similar to the formula for divided difference operators — except the differences are slightly different. This is called the **isobaric divided difference** or the **Demazure operator**.

Now let's define these things carefully in the general setting.

Definition 9.9. The **Demazure operators** $D_1, \ldots, D_{n-1}$ act on $\mathbb{C}[x_1, \ldots, x_n]$ by the following formula:

$$D_i : f \mapsto \frac{f - \frac{x_{i+1}}{x_i} s_i(f)}{1 - \frac{x_{i+1}}{x_i}}.$$

This is the same formula except now we have x_i adn x_{i+1} instead of x and y.

Remark 9.10. $D_i(f) = \partial_i(x_i f)$.

This is in some sense a coincidence because they make sense in general settings (of Coxeter groups), and this relationship holds only in some types — it's not generalizable. The point is if you multiply both the numerator and denominator by x_i, then this becomes obvious. But for other types, this isn't true.

The other difference is that ∂_i decreases the degree by 1, and D_i preserve the degree. (This is why they are called *isobaric*.)

Another nice thing is that these operators *also* satisfy relations very similar to the Coxeter relations:

Lemma 9.11

The D_i satisfy the relations

(1) $D_i^2 = D_i$

(2) $D_i D_{i+1} D_i = D_{i+1} D_i D_{i+1}$

(3) $D_i Dj = D_j D_i$ for $|i - j| \geq 2$.

You can see this from the expression — D_i is trying to symmetrize things, but if you try to symmetrize two things, you don't get anythign else — symmetrizing twice is the same as symmetrizing once. These relations also haev a special name — you have Coxeter relations, nil Coxeter relations, adn now **0-Hecke relations**.

The good thing for us is that hte last two are exactly teh same as Coxeter relations, so we can define Demazure operators not just for simple tranpsositions but for any permutaiton :

Definition 9.12. For any permutation $w \in S_n$, $D_w = D_{i_1} D_{i_2} \cdots D_{i_\ell}$ for any reduced decomposition $w = s_{i_1} s_{i_2} \cdots s_{i_\ell}$.

Now we are ready to state the third formla for Schur polynomials:

Definition 9.13 (Schur as Demazure characters). $s_\lambda^{\text{Dem}}(x_1, \ldots, x_n) = D_{w_0}(x^\lambda)$.

So you try to symmetrize this monomial, but in steps — first wrt one pair of adjacent variables, then another, and so on, until you've completely symmetrized it and you're done.

Theorem 9.14

These four definitions define the same set of polynomials —

$$s_\lambda^{\text{comb}} = s_\lambda^{\text{class}} = s_\lambda^{\text{Schub}} = s_\lambda^{\text{Dem}}.$$

Exercise 9.15. Take some λ and calculate the Dem thing.

§10 September 28, 2022

Theorem 10.1 (Demazure character formula)

For a partition $\lambda = (\lambda_1, \lambda_2, \ldots)$, $s_\lambda(x_1, \ldots, x_n) = D_{w_0}(x^\lambda)$.

Example 10.2

$D_1 \colon x_1 \mapsto x_1 + x_2$. But $D_1 \colon x_1 + x_2 \mapsto x_1 + x_2$ — if something's already symmetric then it does nothing. SO by linearity, $x_2 \mapsto 0$.

Exercise 10.3. What happens if you apply D_{w_0} to *any* monomial $(x_1^{\beta_1} \ldots x_n^{\beta_n})$ (that is not necessarily a partition)?

This will be $\pm$ a Schur polynomial or 0.

Example 10.4

Take $n = 3$, and $\lambda = (4, 2, 0) = \ydiagram{4,2}$.

We want to find $s_\lambda(x_1, x_2, x_3) = \sum \cdots x_1^a x_2^b x_3^c$, with some coefficients. We know $a + b + c$ should equal the sum of all parts of λ; so all monomials that appear here should correspond to lattice points on the plane.

We are going to arrange these lattice points on the plane as follows. Draw the plane where all a, b, c live. Suppose $(4, 2, 0)$ is at a point. Then we can reflect things with respect to the lines: one line is the reflection

s_1, sending it to $(2,4,0)$; another is s_2, giving $(4,0,2)$. We get 6 permutations of $(4,2,0)$ — going clockwise $(4,2,0)$, $(4,0,2)$, $(2,0,4)$, $(0,2,4)$, $(0,4,2)$, $(2,4,0)$. All these points correspond to monomials that appear in Scur polynomials

Then between them, we have their averages $(4,1,1)$, $(3,3,0)$, $(1,4,1)$, $(0,3,3)$, $(1,1,4)$, $(3,0,3)$. Then there will be some other points in between — $(3,2,1)$ and its six permutations $(3,1,2)$, $(2,1,3)$, $(1,2,3)$, $(1,3,2)$, $(2,3,1)$. In the middle we have $(2,2,2)$.

These points represent all possible monomials that might appear in the Schur polynomial, arranged in a hexagonal pattern.

Now let's start with this hexagon (that has 3, 4, 5, 4, 3 points). We're trying to calculate $D_1D_2D_1(x_1^4x_2^2x_3^0)$.

Initially, we just have the single monomial $(4,2,0)$ in the top-right corner. So draw that single point in red with a 1.

Now we first apply D_1, meaning we symmetrize with respect to reflection over the vertical line s_1. What D_1 does is that it takes this single red point and replaces it by an interval of red points in the top row. So now we have the entire top row, all with 1's — this represents the sum of three monomials with coefficients 1, 1, 1.

Now we apply D_2. That means we take this picture, and then symmetrize with respect to reflections around S_2 — the top-right to bottom-left slanted line. We take each of these points and replace them by the whole interval. So now we get the entire half of the hexagon (including the middle line) on the top-right. All of these have coefficients 1. So we started with a point, then got a line segment, then a whole trapezoid (all with coefficients 1).

Then we apply D_1 again. This means we again symmetrize with respect to the vertical line. So the bottom point produces the 3 points in the bottom row. In the second, you get one sum of all 4 things, and one sum of 2. So you get coefficients 1, 2, 2, 1.

In the top row, you get 1, 1, 1. That's already symmetric, so it just remains — we don't need to do anything. Then we have 0, 1, 1, 1. The two points in the middle are already symmetric so they remain the same, and we get 1221. Finally, in the middle row the first 1 gives you a bunch of 1s, the second gives you another three, and the thing in the middle stays. SO we get 1, 2, 3, 2, 1.

So we get a whole hexagon of points with these multiplicities, and that represents $s_{(4,2)}(x_1,x_2,x_3) = x_1^4x_2^2x_3^0 + \cdots + x_1^4x_2x_3 + \cdots + 2x_1^3x_2^2x_1 + \cdots + 3x_1^2x_2^2x_3^2 + \cdots$ — all these numbers represent all possible things. These are the Kostke numbers (the coefficients).

For 3 variables, we can do this. With 4 variables we'd need 3-variable pictures.

Let's check that htese are Kostke numbers. This says $K_{(4,2,0),(2,2,2)}$ — the number of SYT of shape $(4,2,0)$ and weight $(2,2,2)$ — meaning we havetwo 1's, two 2's, two 3's. We can have

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

So then this $K = 3$.

Before we prove this formula, let's make some observations.

- All monomials correspond to points of some hexagon. In general, some polytope. This polytope has a name (because it's related to permutations),

Student Question. What happens if you start with a 1 somehwere else?

Let's start with 1 at the top-right. Then you reflect with respect to this line, and you get all points between λ adn $s_1(\lambda)$ in a closed interval.

If your initial point was in a different area, suppose it was instead on the top-left. Call that thing β. Then when we reflect we get β to $s_1(\beta)$. Now we have to take all integer points in this open interval — we take closed intervals when we go + to −, and open intervals when we go − to +. And the coefficients will be −1's. This is related to *Erhart reciprocity.*

As we can see in this example, everything lives inside this hexagon. More generally, this thing is called the **permutohedron** (or *permutahedron*).

Definition 10.5. For $\lambda \in \mathbb{R}^n$, the **permutohedron** is the polytope

$$\Pi(\lambda) := \operatorname{conv}(\lambda_{w(1)}, \ldots, \lambda_{w(n)}).$$

The convex hull of all points obtained by permuting coordinates of λ.

On the plane, this thing typically looks like a hexagon. THey're in general not necessarily regular; for example, take a point fairly close to the vertical axis. If your point is *on* one of the lines you'll get a triangle.

Fix $\lambda = (\lambda_1, \ldots, \lambda_n)$ to be a partition.

Note that $s_\lambda(x_1, \ldots, x_n) = \sum_{\beta \in \mathbb{Z}^n} K_{\lambda\beta} x^\beta = \sum_{\beta \in \Pi(\lambda) \cup \mathbb{Z}^n} K_{\lambda\beta} x^\beta$. (The firs tis the definition, the second states that only these terms show up:)

Theorem 10.6

$K_{\lambda\beta} \neq 0$ iff $\beta \in \Pi(\lambda) \cup \mathbb{Z}^n$ (i.e. β is a lattice point of the permutahedron).

There is a nother well-known characterization of nonzero Kostke numbers. A related result, as we saw last time using BK, we saw that $K_{\lambda\beta} = K_{\lambda\tilde{\beta}}$ for any permutation $\tilde{\beta}$ of β. So to characterize these things, we can assume β is not just an arbitrary vector, but a partition. To emphasize this, we will denote it by μ — assume $\mu = (\mu_1 \geq \cdots)$ is a partition.

Theorem 10.7

$K_{\lambda\mu} \neq 0$ if and only if $\lambda_1 \geq \mu_1$, $\lambda_1 + \lambda_2 \geq \mu_1 + \mu_2$, and so on — any partial sum of parts of λ should be at least the corresponding partial sum of μ. Finally, we should also have $|\lambda| = |\mu|$.

This gives a certain partial order on partitions — if λ and μ satisfy these relations we say $\lambda \geq \mu$. This partial order is called the **dominance order** on partitions. Then Kostke numbers $K_{\lambda\mu}$ are nonzero iff $\lambda \geq \mu$ in the dominance order.

These two theorems are equivalent, but to see that we need a different result. In genreal, there's two ways to define a polytope. One is to give a bunch of points, and say the polytope is the convex hull of those points (that's how we defined the permutahedron). Dually, we can define polytopes by giving a bunch of inequalities. Rado's theorem relates these two descriptiosn for the permutahedron.

Theorem 10.8 (Rado's Theorem)

$\Pi(\lambda)$ is the set of $(x_1, \ldots, x_n) \in \mathbb{R}^n$ that satisfy the following inequalities:

- $x_i \leq \lambda_1$.
- $x_{i_1} + x_{i_2} \leq \lambda_1 + \lambda_2$ for all $i_1 \neq i_2$.
- In general, $x_{i_1} + \cdots + x_{i_k} \leq \lambda_1 + \lambda_2 + \cdots + \lambda_k$ for any k and distinct indices $i_1, \ldots, i_k$.

And the condition that $x_1 + \cdots + x_n = \lambda_1 + \cdots + \lambda_n$.

This will be left as an exercise. If you believe it, it is easy to see that the things are equivalent.

Now let's prove the formula for Schur polynomials in terms of Demazure operators. Last time, we gave a formula for Schur polynomials as Schubert polynomials — $\partial_{w_0}(x^{\lambda+\delta})$ where $\delta = (n-1, n-2, \ldots, 1, 0)$. And meanwhile this should equal $D_{w_0}(x^\lambda)$.

First let's introduce some notatin. Let X_i be the operator $f \mapsto x_i f$ acting on polynomials. In particlular $X^\delta = X_1^{\delta_1} X_2^{\delta_2} \cdots$.

Theorem 10.9

$D_{w_0} = \partial_{w_0} X^\delta$ (this denotes the composition of operators on $\mathbb{C}[x_1, \ldots, x_n]$).

Example 10.10

Let $n = 3$. Then

$$D_{w_0} = D_1 D_2 D_1.$$

But we know $D_i = \partial_i X_i$, so then

$$D_{w_0} = \partial_1 X_1 \partial_2 X_2 \partial_1 X_1.$$

On the RHS, we get

$$\partial_1 \partial_2 \partial_1 X_1^2 X_2.$$

First to see why this is nontrivial, note that these are operators, adn sometimes the ∂s commute with the X_I and sometimes they are not. In fact, $\partial_i X_j = X_j \partial_i$ if $j \neq i, i+1$. (This is because ∂_i affects x_i adn x_{i+1}, so it commutes with anything else.)

We wnat to somehwo move all X's in the right in our D formula. Sometimes you can do this easily — X_1 and ∂_2 commute, giving

$$\partial_1 \partial_2 X_1 X_2 \partial_1 X_1.$$

But now you are stuck because X_1 and X_2 don't commute with ∂_1. But in fact $X_1 X_2$ *does* commute with ∂_1 — it's symmetric with respect to permutations of x_1 and x_2, so we can move this product through ∂_1. Now we get what we wanted, which is

$$\partial_1 \partial_2 \partial_1 X_1 X_2 X_1.$$

This gives the idea for how to do it — write a nice reduced decomposition and somehow move X's thorugh teh ∂'s, but sometimes you have to arrange several X's together. If you do this in a smart way then you can always do this.

§11 September 30, 2022

§11.1 Operators on Polynomials

Last time, we discussed some operators on $\mathbb{C}[x_1, \dots, x_n]$ – the divided difference operator $\partial_i \colon f \mapsto \frac{1}{x_i - x_{i+1}}(1 - s_i)f$, and the Demazure operator (or isobaric divided difference operator) $D_i \colon f \mapsto \frac{1}{1 - x_{i+1}/x_i} \cdot (1 - \frac{x_{i+1}}{x_i} s_i)(f)$. We also had the simple operator $X_i \colon f \mapsto x_i f$. We can immediately notice that

$$D_i = \partial_i X_i.$$

In other words, to get D_i, we first multiply our polynomial by x_i and then apply ∂_i.

Furthermore, we defined these operators not only for simple transpositions, but for *any* permutation w — given any $w \in S_n$ we can pick a reduced decomposition $w = s_{i_1} \cdots s_{i_\ell}$, and then define

$$\partial_w := \partial_{i_1} \cdots \partial_{i_\ell}$$

and

$$D_w = D_{i_1} \cdots D_{i_\ell}.$$

For any vector $\beta = (\beta_1, \dots, \beta_n)$ of nonnegative integers, we also define $x^\beta = x_1^{\beta_1} \cdots x_n^{\beta_n}$.

Student Question. Is there an invariant definition of ∂_w or D_w — one that doesn't depend on the reduced decomposition?

This is the simplest way; but for the longest permutation, there is an invariant description, which will be left for us as an exercise — for the longest permutation,

$$\partial_{w_0} \colon f \mapsto \frac{1}{\prod_{i<j} x_i - x_j} \cdot \sum_{w \in S_n} (-1)^{\ell(w)} w(f).$$

So this is true for the longest permutation, at least. Whether there exists one for others — we will talk about this more later, when we discuss Schubert polynomials.

Theorem 11.1

$D_{w_0} = \partial_{w_0} X^\delta$, where δ is the staircase partition $(n-1, n-2, \dots, 0)$.

This theorem immediately implies that our two definitions in terms of divided difference operators and Demazure operators are equivalent.

Proof. We start with a simple lemma:

Lemma 11.2

First, ∂_i commutes with X_j if $j \neq i, i+1$.

Second, ∂_i commutes with any polynomial $f(X_1, \dots, X_n)$ that is invariant under switching X_i and X_{i+1} — in other words, such that $s_i(f) = f$.

This is clear, since ∂_i only switches around x_i and x_{i+1}.

Example 11.3

∂_1 doesn't commute with X_1, but it does commute with $X_1 X_2$ or $X_1 + X_2$.

Now in order to prove this, we want to take a particularly nice reduced word decomposition of w_0:

$$w_0 = (s_1 s_2 s_3 \cdots s_{n-1})(s_1 s_2 s_3 \cdots s_{n-2}) \cdots (s_1 s_2)(s_1).$$

Note that $s_1 s_2 \cdots s_{n-1}$ is the cycle $(1, 2, 3, \dots, n)$ (sending $1 \mapsto 2$, $2 \mapsto 3$, and so on), the second term is $(1, 2, 3, \dots, n-1)$, and so on; $s_1 s_2 = (1, 2, 3)$ and $s_1 = (1, 2)$. We can check that the product of these cycles will give us the longest permutation. For example, in terms of wiring diagrams, this looks like:

Label the wires 1, 2, 3, ..., n from the bottom. To write a wiring diragram, we read from the right. So we first take the second wire and send it all the way down, then the third wire and send it all the way down, then the next, and so on. (Lines are straight right, then 45° down,t hen 45° up.)

Now we can write

$$D_{w_0} = (\partial_1 X_1 \partial_2 X_2 \cdots \partial_{n-1} X_{n-1})(\partial_1 X_1 \cdots \partial_{n-2} X_{n-2}) \cdots (\partial_1 X_1 \partial_2 X_2)(\partial_1 X_1).$$

Looking at the first expression, we can see that we can move X_1 all the way to the right in that expression, and we can do the same in the next one, and so on — so then we can write

$$D_{w_0} = (\partial_1 \partial_2 \cdots \partial_{n-1}) X_1 \cdots X_{n-1} \cdot (\partial_1 \partial_2 \cdots \partial_{n-2}) X_1 \cdots X_{n-2} \cdots \partial_1 X_1.$$

We cannot individually commute all these X's through the next ∂'s. But we can combine them together – $X_1 X_2 \cdots X_{n-1}$ is symmetric in the first $n-1$ variables, so then it commutes with everything else, and we can move them all the way to the end. Then we can combine the next block and move them all the way to the end, and so on. So we get

$$\partial_{w_0} \cdot X^\delta.$$

So somehow we were able to commute all X's to teh right. □

Looking at this proof, the reason it works is that we were able to find a very special reduced decomposition. If you pick some other reduced decomposition, it may or may not work — you may not be able to move all X's to the right.

Some observations:

- $D_1 = \partial_1 X_1$, $D_2 = \partial_2 X_2$;
- $D_{w_0} = \partial_{w_0} X^\delta$;
- $D_{s_1 s_2} = D_1 D_2 = \partial_1 X_1 \partial_2 X_2 = \partial_1 \partial_2 X_1 X_2$.

In all these examples, it looks like we can commute all X's to the right.

Question 11.4. Does this always happen?

Let's try to do this for $D_{s_2 s_1} = \partial_2 X_2 \cdot \partial_1 X_1$. We are not allowed to move X_2 through ∂_1, and we can actually check that this is *not* of the form $\partial_2 \partial_1 X^\beta$. So this permutation is somehow not good — for the other permutations on the list we could do this method, but not for *all* permutations.

Exercise 11.5. Characterize the permutations $w \in S_n$ such that $D_w = \partial_w \cdot X^\beta$ for some nonnegative integer vector β depending on w. Also, explicitly express β as a function of w.

As we have seen, for $n = 3$ we get all but one.

Example 11.6

For S_3, we have dix permutations.

$$\beta(w_0) = (2,1,0)$$
$$\beta(s_1 s_2) = (1,1,0)$$

$$\beta(s_1) = (1,0,0) \quad \beta(s_2) = (0,1,0)$$
$$\beta(id) = (0,0,0)$$

As a hint, there is something called the **Lehmer code** of a permutation. Let $\text{code}(w) = (c_1, \ldots, c_n)$ for a permutation $w \in S_n$. This is defined as follows:

Definition 11.7. $c_i = \#\{j > i \mid w(i) < w(j)\}$.

Theorem 11.8

The Lehmer code gives a bijection between all permutations and all nonnegative integer vectors where the first entry is at most $n-1$, the second entry at most $n-2$, and so on.

in other words, $\text{code}(w) \leq \delta$.

Observe that in this example, β is exactly the code of the corresponding permutation.

Your problem is to figure out for which operations $D_w = \partial_w X^{\text{code}}$.

Student Question. Why do we write β_n if it's always 0?

Because it is easier to write n than $n-1$. You can ignore it. But traditionally people write n for convenience.

In general, if you figure out how to solve this problem, you can think about another thing:

Exercise 11.9. We know $D_w = \partial_w X^{\text{code}(w)} + \text{something else in general}$. What can we say about the other terms?

The reason it works for w_0 is that we were able to find a nice decomposition. What kinds of features of the reduced decomposition should you have for this argument to work?

§11.2 Definitions of Schur Polynomials

Earlier, we've seen many definitions of Schur polynomials:

Theorem 11.10

$s_\lambda^{\text{comb}} = s_\lambda^{\text{class}} = s_\lambda^{\text{Schub}} = s_\lambda^{\text{Dem}}$.

We just proved the last equality. The middle equality follows from the lemma about the explicit form for ∂_{w_0} left as an exercise.

So now it remains to rpove the first. The easiest way to see this is to use the theory of symmetric functions.

§11.3 SYmmetric Functions

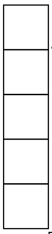
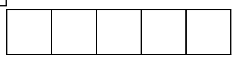
We will now let Λ be the ring of symmetric functions in *infinitely* many variables x_1, x_2, x_3, $\ldots$. There are several well-known classes of symmetric functions:

Definition 11.11. The **elementary symmetric functions** are

$$e_k := \sum_{i_1 < \cdots < i_k} x_{i_1} x_{i_2} \cdots x_{i_k}.$$

Definition 11.12. The **complete homogeneous symmetric functions** are

$$h_k := \sum_{j_1 \leq j_2 \leq \cdots \leq j_k} x_{j_1} x_{j_2} \cdots x_{j_k}.$$

In other words, e_k is the sum of all square-free monomials of degree k, and h_k the sum of *all* monomials of degree k. We can easily notice that e_k adn h_k are special classes of combinatorially defined Schur functions: $e_k = s_{1^k}$, where 1^k is a column . Then SSYT are defined by putting in $i_1 < i_2 < \cdots$. SImilarly, h_k is the Schru function for a single *row* —  where we input $j_1 \leq j_2 \leq \cdots \leq j_n$.

Theorem 11.13 (Fundamental Theorem on Symmetric Functions)

Two equivalent parts:

- $\Lambda = \mathbb{C}[e_1, e_2, \ldots]$ — in other words, any symmetric function can be written uniquely in terms of the elementary ones.
- $\Lambda = \mathbb{C}[h_1, h_2, \ldots]$.

Before we prove this, let's see why (1) and (2) are equivalent — any complete homogeneous can be expressed in terms of elemnetary, and the other way around.

We can easily do this by using the following formula:

Lemma 11.14

$e_0 = h_0 = 1$.

$e_1 h_0 - e_0 h_1 = 0$. (This means $e_1 = h_1$, which is obvious.)

$e_2 h_0 - e_1 h_1 + e_0 h_2 = 0$.

And so on — in general

$$\sum_{k=0}^{n} (-1)^k e_k h_{n-k} = \begin{cases} 1 & n = 0 \\ 0 & n > 0 \end{cases}.$$

Then we can easily express either in terms of hte other by induction — we know $h_1 = e_1$, then moving terms to one side and moving everything else to the other gives h_2 in terms of stuff we've alrady expressed, and so on.

This is a class in combinatorics, so let's see a nice combinatorial way to prove this formula. This is a method used for many problems in combinatorics, called a *sign-reversing involution*. We want to see that this formula holds for these polynomials:

$$\sum_{k=0}^{n}(-1)^k e_k h_{n-k} = \sum_{k=0}^{n}(-1)^k \sum_{i_1<\cdots<i_k} x_{i_1}\cdots x_{i_k} \cdot \sum_{j_1\geq j_2\geq\cdots\geq j_{n-k}} x_{j_1}\cdots x_{j_{n-K}}.$$

(we wrote h in terms of weakly decreasing instead of weakly increasing — of course that's the same thing, but this will be helpful).

We want to show that all terms on teh RHS will cancel each toher — in other words, we want ani nvolution on pairs of i's and j's that swaps sign. So we want a *sign-reversing involution* on the set of pairs

$$\{(i_1,\ldots,i_k)\text{strictly increasing}, (j_1,\ldots,j_k)\text{weakly decreasing}\}.$$

What we want to do is change the parity of k — increase or decrease by 1.

Take our involution $\phi\colon ((i_1,\ldots,i_k),(j_1,\ldots,j_{n-k}))\mapsto((i_1,\ldots,i_k,j_1),(\ldots))$ or $(i_1,\ldots,i_{k-1}),(i_k,j_1,\ldots)$. We can do the first if $i_k<j_1$, and the second if $i_k\geq j_1$.

We conclude that the thing equals 0 if $n\geq 1$.

From this we can deduce that the two parts of the fundamental theorem are equivalent to each other. Now let's prove it.

There is *another* collection of symmetric functions:

Definition 11.15. The **monomial** symmetric functions are labeled by partitions $\lambda=(\lambda_1,\ldots,\lambda_\ell)$. $m_\lambda = x_1^{\lambda_1}\cdots x_\ell^{\lambda_\ell}$ + all other monomials obtained by permutations of variables.

Clearly m_λ is symmetric, and $\{m_\lambda\}$ is a linear basis of λ. (By definition, a symmetric function is a combination of monomials so that any two monomials obtained by permutations of coordinates have the same coefficients. You can then combine these.)

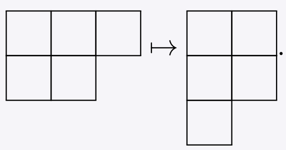
If $\lambda_1=\lambda_2$, we still have coefficient 1 — you only count a monomial once.

We can also define $e_\lambda = e_{\lambda_1}e_{\lambda_2}\cdots e_{\lambda_\ell}$, and $h_\lambda = h_{\lambda_1}\cdots h_{\lambda_\ell}$. Then the fundamental theorem of symmetric functions is equivalent to saying that $\{e_\lambda\}$ is a linear basis of Λ, and the second part to saying that $\{h_\lambda\}$ is a linear basis of Λ.

So what we need to show is that you can express any monomial symmetric function as a linear combination of e_λ, and vice versa. These are related to each toher by a triangular transformation — an upper triangular matrix with 1's on the diagonal. But in order to talk about such things — this will be a matrix with rows and columns labelled by *partitions* — we need to order partitions. We can do that lexicographically:

Fix n, and take the lexicographical ordering on all partitions $\lambda\vdash n$. In other words, $\lambda >_{\text{lex}} \mu$ if $\lambda_1>\mu_1$, or $\lambda_1=\mu_1$ and $\lambda_2>\mu_2$, or so on — if in the first place they differ, $\lambda_i>\mu_i$.

Definition 11.16 (Conjugate Partitions)**.** For a partition λ, its **conjugate partition** λ' is the partition whose Young diagram is obtained by transposing the Young diagrma of λ. For example



Lemma 11.17

$e_\lambda = m_{\lambda'} + \sum_{\mu<\lambda'} a_{\lambda\mu} m_\mu$ for some coefficients $a_{\lambda\mu}$.

This follows immediately from definitions, but you should see why we get conjugate partitions.

Also, instead of lexicographical order you can take any linear ordering compatible with dominance ordering (any linear extension of the dominance order).

§12 October 3, 2022

§12.1 Symmetric Functions

Last time, we stated some definitions:

- m_λ are the monomial symmetric functions (we take one monomial $x_1^{\lambda_1} \cdots x_n^{\lambda_n}$ and permute variables in all ways).
- $s_\lambda = s_\lambda^{\text{comb}}$ are the Schur functions (we'll use the combinatorial definition);
- $e_\lambda = e_{\lambda_1} \cdots e_{\lambda_n}$ where e_k are the elementary symmetric functions;
- $h_\lambda = h_{\lambda_1} \cdots h_{\lambda_n}$ where h_k are the complete homogeneous symmetric functions.

There is another important class, the *power* symmetric functions, which we will discuss later.

Theorem 12.1

$\{m_\lambda\}$, $\{s_\lambda\}$, $\{e_\lambda\}$, and $\{h_\lambda\}$ are linear bases of Λ.

The first claim is obvious — any symmetric function can uniquely be written as a combination of monomial symmetric functions. The two claims about $\{e_\lambda\}$ and $\{h_\lambda\}$ are equivalent to the fundamental theorem of symmetric functions — that $\Lambda = \mathbb{C}[e_1, e_2, \ldots] = \mathbb{C}[h_1, h_2, \ldots]$.

Proof. First, for m_λ this is immediate.

For s_λ, we know that we can write $s_\lambda = \sum_\mu K_{\lambda\mu} m_\mu$ (we defined $K_{\lambda\beta}$ for general vectors β, but now we're combining all terms with the same set of exponents into one term m_μ — $K_{\lambda\mu}$ is the number of SSYT of shape λ and weight μ).

It is easy to see that $K_{\lambda\mu}$ form an upper triangular matrix with 1's on the diagonal — in other words $K_{\lambda\lambda} = 1$, and $K_{\lambda\mu} = 0$ unless $\lambda \geq \mu$ in the lexicographical order. (More precisely, $K_{\lambda\mu}$ is nonzero exactly when $\lambda \geq \mu$ in the dominance order.)

So then the s_λ are related to m_λ by an invertible linear transformation, which means they form a basis as well.

For e_λ, we'll do the same thing — we'll express e_λ as a linear combination of m_λ. We can see that

$$e_\lambda = \sum_\mu c_{\lambda\mu} m_\mu,$$

and we claim that the $c_{\lambda\mu}$ also form an upper triangular matrix with 1's on the diagonal. But here we have to be more careful:

Claim 12.2 — If λ' is the conjugate of λ, then $c_{\lambda\lambda'} = 1$, and $c_{\lambda\mu'} = 0$ unless $\mu \leq \lambda'$ in the lexicographical order.

Again, this becomes an "if and only if" in the dominance order.

Example 12.3

Let's try to express $e_{3,2}$ as a linear combination of monomials. We have

$$e_{3,2} = e_3 e_2 = (x_1x_2x_3 + x_1x_2x_4 + \cdots)(x_1x_2 + x_1x_3 + \cdots).$$

Now we can imagine expanding term-wise and seeing what kinds of monomials we can get. We can get $x_1^2x_2^2x_3$, and any permutations of this monomial, so that gives $m_{2,2,1}$.

If we multiply $x_1x_2x_3 \cdot x_1x_4$, then we get $x_1^2x_2x_3x_4$. If we multiply $x_1x_2x_3$ by x_4x_5, then we get $x_1x_2x_3x_4x_5$. So these are the three types of monomials we can get:

$$e_{3,2} = m_{2,2,1} + am_{2,1,1,1} + bm_{1,1,1,1,1}.$$

The leading one is $(2, 2, 1) = \ydiagram{2,2,1}$, which is exactly conjugate to $\lambda = \ydiagram{3,2}$.

Exercise 12.4. Find an explicit combinatorial rule for the coefficients $c_{\lambda\mu}$.

Now let's do $\{h_\lambda\}$. IF we try the same recipe that we did for e's, we will actually fail — we can still write $h_\lambda = \sum d_{\lambda\mu} m_\mu$, but this is not actually going to give an upper triangular matrix. (If we expand out complete homogeneous functions, *all* the $d_{\lambda\mu}$ will be nonzero for *any* λ and μ with $|\lambda| = |\mu|$.)

So this recipe we did here works for elementary, but not complete homogeneous. SO we need to do something else, and we actually did thsi last time — we can deduce this from the fact that $\{e_\lambda\}$ forms a basis, since we can express any elementary symmetric function in terms of homogeneous ones and so on.

Lemma 12.5

$$\sum_{k=0}^{n} (-1)^k e_k h_{n-k} = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise.} \end{cases}$$

By induction, this means we can express any homogeneous symmetric function in terms of elementary ones, and so on.

We can now try to write this identity in a more compact form. Consider the generating function

$$E(t) = e_0 + e_1 t + e_2 t^2 + \cdots,$$

and similarly

$$H(t) = h_0 + h_1 t + h_2 t^2 + \cdots.$$

Then our lemma, using this notation, can be written as

$$E(t) \cdot H(-t) = 1.$$

Last time, we saw a proof using the involution principle — we constructed a sign-reversing involution — but there is a more trivial proof in terms of generating functions. We can notice that

$$E(t) = \prod_{i=1}^{\infty} (1 + x_i t),$$

since if we expand this infininte product and look at the t^n coefficient, we need to take some subset of x_i with size n, adn then we sum over all possible subsets. Meanwhile, we have

$$H(t) = \prod_{i=1}^{\infty} \frac{1}{1 - x_i t} = \prod_{i=1}^{\infty} (1 + x_i t + x_i t^2 + \cdots),$$

since we get t^n from every monomial of degree n. But from these expressions, it is straightforward to see that $E(t) \cdot H(-t) = 1$.

Now this lemma gives a recursive way to describe the h_k and e_k in terms of each other.

Question 12.6. Can we do this nonrecursively?

The answer is yes. We can rewrite the identity in another way: introduce the infinite matrices

$$H = \begin{bmatrix} 1 & h_1 & 1 & & \\ h_2 & h_1 & 1 & & \\ h_3 & h_2 & h_1 & 1 & \\ h_4 & h_3 & h_2 & h_1 & 1 \end{bmatrix}$$

and

$$E = \begin{bmatrix} 1 & -e_1 & 1 & \\ e_2 & -e_1 & 1 & \\ -e_3 & e_2 & -e_1 & 1 \end{bmatrix},$$

which are lower triangular matrices indexed by positive integers and have the same entries on their diagonal. We could do the same t hing taking these to be $(n+1) \times (n+1)$ matrices instead (ending with h_n, h_{n-1}, $\ldots$). Then our lemma is equivalent to stating that $E = H^{-1}$. This means the entries of E are cofactors of H, and vice versa (both have determinant 1; the cofactor is the determinant obtained by removing a row and column, and dividing by the determinant of the whole matrix — so each h_n is given by a $n \times n$ matrix in the e's and vice versa). More explicitly, we get the expression:

Lemma 12.7

$$e_n = \begin{vmatrix} h_1 & 1 & 0 & \cdots \\ h_2 & h_1 & 1 & \\ h_n & \cdots & h_2 & h_1 \end{vmatrix}.$$

You can do the same thing to get h_n in terms of the e_k — it's the same, but you replace the h's by e's.

Note that we do not need alternating signs here — we should also define $\widetilde{H}$ and $\widetilde{E}$ by using alternating signs for the h_k instead of the e_k, and then we have the fact $\widetilde{H} \cdot \widetilde{E} = 1$ as well.

Now we can see there is some kind of duality between the e's and h's. This gives us an involution on Λ, denoted ω:

Lemma 12.8

There exists a unique algebra homomorphism $\omega \colon \Lambda \to \Lambda$ such that ω sends $e_k \mapsto h_k$ and $h_k \mapsto e_k$ for all k.

Proof. If we remove the second statement, then this is obvious — since the e_k generate Λ. So what we want to show is that this homomorphism sends the h_k back to e_k. But that's exactly what the matrices tell us (because they're the same — to express the h's in terms of e's is the esame as to express the e's in terms of h's). ∎

Note that more generally, this also means ω sends $e_\lambda \mapsto h_\lambda$ and $h_\lambda \mapsto e_\lambda$. □

In fact, we claim ω does nice things to Schur functions as well. For that, we'll use another useful lemma, implied by the fundamental theorem:

Lemma 12.9

Suppose $\{f_\lambda\}$ and $\{g_\lambda\}$ are two collections of symmetric functions, such that:

1. $\deg f_\lambda = \deg g_\lambda = |\lambda|$;
2. $f_\varnothing = g_\varnothing = 1$;
3. We have a formula $e_n f_\lambda = \sum_\mu d_{\lambda\mu n} f_\mu$, and the exact same formula $e_n g_\lambda = \sum_\mu d_{\lambda\mu n} g_\mu$.

Then $\{f_\lambda\}$ and $\{g_\lambda\}$ are linear bases of Λ, and $f_\lambda = g_\lambda$.

The proof is basically immediate from the fundamental theorem.

Proof. First, for why they're both linea rbases, we want to use the given formula to see that f_λ forms a linear basis. Let's assume f_λ is the empty partition; then $e_n \cdot 1$ is some linear combination of f_μ. Then we multiply by another e_k and apply this again to simplify. Then we can express any e_n as a linear combination of f_μ's, so they span the space of symmetric functions. Meanwhile, to see that they form a baiss, you can see that they're labelled by partitions, and the dimension of the space of symmetric functions of degree n is given by the partition function.

Then we get the same expression for the e's in terms of f's and g's, which means that the two are the same. □

In other words, if you have some collection of symmetric funciton sthat's a candidate for a basis, then all you need to check is that the rule is what you expect.

But in particular, we can apply this lemma to show that the combinatorial and classical definitions of Schur functions are the same — it's enough to check that the same rule holds for elementary symmetric functions, in this way.

These rules are well-known.

Theorem 12.10 (Pieri Rules)

The following rules hold for both s_λ^{comb} and s_λ^{class}:

- $e_n s_\lambda = \sum_\mu s_\mu$, where the sum is over all μ such that $\mu - \lambda$ is a **vertical n-strip** — this means we have some λ, and we add some boxes to λ such that we add exactly n boes and these form a disjoint union of several vertical strips. So $\mu - \lambda$ has n boxes, and any row contains at most one box.
- Similarly, we have $h_n s_\lambda = \sum_\mu s_\mu$, where now the sum is over μ such that $\mu - \lambda$ is a *horizontal* n-strip — we have a partition λ and add n boxes such that in each *column* there's at most one box.

It's enough to just check one of these identities for both combinatorial and classically defined Schur functions, to show that they are the same.

For combinatorial Schur functions, both can be checked usign RSK. (This will be left as an exercise on the problem set.)

For classically defined Schur functions, as the quotient of two determinants $a_{\lambda+\delta}/a_\delta$, we can also directly check (1) — if we play around with determinants, this is not hard to see from the definition. Interestingly,

(2) is hard to prove using the determinants. SO for some reason, elementary symmetric functions are easier to work with. BUt once ew've proven (1), that automatically immplies (2). (2) is harder to prove for classically defined, but it's as easy as (1) for combinatorially defined, and so we can prove both identities for comb and (1) for class, and use the lemma to show class and comb are the same, and that implies (2) for class as well.

Now for the second half of the lecture (it is 1:54), we will see three proofs of the following theorem:

Theorem 12.11

$\omega\colon s_\lambda \mapsto s_{\lambda'}$.

Proof 1. We can compare the two formulas in the Pieri rule — we can get one formula from the other by conjugating λ (which swaps horizontal and vertical strips), and using the lemma to show that this is well-defined. □

Proof 2. Use Cauchy identities. There are two — the usual Cauchy identity and the dual one, related to dual RSK. We will see these next class. □

Proof 3. Clearly s_λ and $s_{\lambda'}$ are bases, so we can define ω by this and check that it's a ring homomorphism.

There is a formula for the product of Schur functions called the **littlewood insertion rule**,

$$s_\lambda \cdot s_\mu = \sum c^\nu_{\lambda\mu} s_\nu.$$

Then to prove this, we want to show that $c^\nu_{\lambda\mu}$ equals the same thing when we conjugate all the relevant partitions. There is a way to describe these coefficients that makes the symmetry clear. □

§13 October 5, 2022

§13.1 Specialization of Schur Functions

Example 13.1

What is $s_\lambda(1, 1, \dots, 1)$?

We assume $\lambda = (\lambda_1, \dots, \lambda_n)$ has at most n parts — if λ has more than n parts, then s_λ must be 0. By the combinatorial definition, $s_\lambda(1, 1, \dots, 1)$ is the number of SSYT of shape λ with entries between 1 and n — or equivalently, Gelfand–Tsetlin patterns with top row $\lambda_1, \dots, \lambda_n$.

Recall that GT patterns are triangular arrays of numbers

$$\begin{array}{ccccccc} \lambda_1 & & \lambda_2 & & \cdots & & \lambda_n \\ & \mu_1 & & \mu_2 & & \cdots & \end{array},$$

where entries are weakly increasing right to left.

There is a general formula for this:

Theorem 13.2 (Weyl's Dimension Formula)

$$s_\lambda(1, 1, \dots, 1) = \prod_{1 \le i < j \le n} \frac{\lambda_i - \lambda_j + j - i}{j - i}.$$

Weyl's dimension formula is actually more general than this. The original reason people studied SSYT comes from representation theory — SSYT correspond to a basis of irreducible representations of SL_n, adn in particular $s_\lambda(1, 1, \ldots, 1) = \dim V_\lambda$, where V_λ is an irreducible representation of SL_n. There is a general formula for semisimple Lie groups, and for SL_n it specializes to the above theorem.

This may vaguely remind you of the classical definition of Schur functions as the quotient of two determinants. In fact, actually it is possible to deduce this formula from the classical definition.

Exercise 13.3. Deduce this formula from the definition s_λ^{class}.

Intrestingly, there is another formula for the same number. Recall that there is a hook-length formula for SYT. Here we're counting different things, but there's a similar result:

Theorem 13.4 (Stanley's Hook-Content Formula)
We have

$$s_\lambda(1, 1, \ldots, 1) = \prod_{(i,j)\in\lambda} \frac{n + c(i,j)}{h_{i,j}},$$

where $h(i,j)$ is the *hooklength* of box (i,j) and $c(i,j) = j - i$ is the **content** of box (i,j).

This is the same denominator, but the numerator is different.

Example 13.5
Take $n = 3$, and $\lambda = (3, 2, 0)$, so

$$\lambda = \begin{array}{|c|c|c|}\hline & & \\ \hline & \\ \cline{1-2}\end{array}.$$

Then from Weyl we have

$$s_{\begin{array}{|c|c|c|}\hline & & \\ \hline & \\ \cline{1-2}\end{array}}(1, 1, 1) = \frac{3-2+1}{1} \cdot \frac{3-0+2}{2} \cdot \frac{2-0+1}{1} = 15.$$

Meanwhile Stanley gives: the hooklengths are

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

and the contents are

$$\begin{array}{|c|c|c|}\hline 0 & 1 & 2 \\ \hline -1 & 0 \\ \cline{1-2}\end{array},$$

so then we take

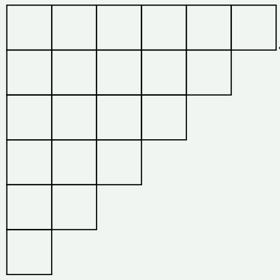
$$\frac{(3+0)(3+1)(3+2)(3-1)(3+0)}{4 \cdot 3 \cdot 1 \cdot 2 \cdot 1} = 15.$$

Which formula is easier? It depends — Weyl's is the product of $\binom{n}{2}$ terms, and Stanley's is the number of boxes in the Young diagram. If n is small and the Young diagram is large (horizontally), then Weyl's is better. On the other hand, if our Young diagram is fixed so $|\lambda|$ is not very large, but n is very large, then Stanley's will be more efficient. So it depends on the region where we're trying to calculate this expression.

Exercise 13.6. Prove combinatorially that the two formulas are equivalent.

Example 13.7

Suppose that we have the staircase partition

$$\lambda = (n-1, n-2, \dots, 1, 0) =$$ .

Now according to Weyl

$$s_\lambda(\underbrace{1, 1, \dots, 1}_{n}) = \prod_{i<j} \frac{j - i + j - i}{j - i}.$$

Meanwhile according to Stanley's formula, we get

$$s_\lambda(1, 1, \dots, 1) = \prod_{i+j<n} \frac{n + j - i}{2(n - (j+i)) + 1}.$$

(We can see that all hook lengths are odd.) These expressions are pretty different, but somehow you can show that both of these things are equal. From the first, it's easy to see that the answer is $2^{\binom{n}{2}}$. So somehow, the number of SSYT of this shape is a power of 2.

Exercise 13.8. Prove this bijectively.

Example 13.9

FOr $n = 3$, all Gelfand–Tsetlin patterns with top row $(2, 1, 0)$ are:

$$\begin{array}{ccc} 2 & 1 & 0 \\ 2 & 1 & \\ 2 & & \end{array}, \begin{array}{ccc} 2 & 1 & 1 \\ 2 & 1 & \\ 1 & & \end{array}, \begin{array}{ccc} 2 & 1 & 0 \\ 2 & 0 & \\ 2 & & \end{array}, \begin{array}{ccc} 2 & 1 & 0 \\ 2 & 0 & \\ 1 & & \end{array}, \begin{array}{ccc} 2 & 1 & 0 \\ 2 & 0 & \\ 0 & & \end{array}$$

and

$$\begin{array}{ccc} 2 & 1 & 0 \\ 1 & 1 & \\ 1 & & \end{array}, \begin{array}{ccc} 2 & 1 & 0 \\ 1 & 0 & \\ 1 & & \end{array}, \begin{array}{ccc} 2 & 1 & 0 \\ 1 & 0 & \\ 0 & & \end{array}.$$

Recall that the Gelfand-Tsetlin polytope $T(\lambda) \subseteq \mathbb{R}^{\binom{n}{2}}$ is the set of all points which are real-valued GT patterns with top row λ. In other words, we have a bunch of inequalities.

Example 13.10

For $n = 3$, $GT(\lambda_1, \lambda_2, \lambda_3)$ is the set

$$(x, y, z) \in \mathbb{R}^3 \mid \lambda_1 \geq x \geq \lambda_2 \geq y \geq \lambda_3, x \geq z \geq y.$$

We can ask some questions about this polytope —

Question 13.11. What is the number of lattice points of this polytope?

In other words, we want to find $\#GT(\lambda) \cap \mathbb{Z}^{\binom{n}{2}}$. This is just $s_\lambda(1,1,1)$, and we have two formulas for it:

$$\prod_{i<j} \frac{\lambda_i - \lambda_j + j - i}{j - i}.$$

Question 13.12. What is the volume of this polytope?

If we know how to calculate the number of lattice points, we can actually calculate the volume. You can *define* volume in terms of counting lattice points — dilate the polytope, count the number of lattice points, and take the leading term. So by definition

$$\text{volume} = \lim_{t \to \infty} \frac{\text{lattice points in } GT(t \cdot \lambda)}{t^{\binom{n}{2}}}.$$

We can look at either formula and see what happens when we dilate the polytope. When we dilate, Weyl will behave better, because the number of terms will be fixed (while when we use Stanley's, we get more and more boxes, so more and more terms). So then

$$\text{Vol} = \lim_{t \to \infty} \prod_{i<j} \frac{t(\lambda_i - \lambda_j) + j - i}{j - i} \cdot \frac{1}{t^{\binom{n}{2}}}.$$

In the limit, we can suppress the ocnstnat and only care about $\lambda_i - \lambda_j$. So we just get

$$\text{Vol} = \prod_{i<j} \frac{\lambda_i - \lambda_j}{j - i}.$$

So this is even simpler than the number of lattice points.

Now we have a nice expression for the volume. Ehrhart theory studies polynomials of this type.

§13.2 Ehrhart Theory

Let P be a lattice polytope in $\mathbb{R}^N$ (so all vertices are lattice points).

Theorem 13.13

There exists a unique polynomial $L_P(t) \in \mathbb{R}[t]$, called the **Ehrhart polynomial**, such that for all $t \in \mathbb{Z}_{\geq 0}$, $L_P(t)$ equals the number of lattice points in the dilated polytope tP.

One method of proving this is to check it for a simplex. If you start dilating the standard simplex, then the number of lattice points are given by a binomial coefficient, which is a polynomial in t. Then you can triangulate any lattice polytope into simplices, and from this you can calculate the Ehrhart polynomial of any polytope.

Question 13.14. Does $L_P(t)$ have integer coefficients? What about positive coefficients?

In general, the answer to both questions are no — there are counterexamples to each. For example, for the standard simplex, the Erhart polynomial will be $n + \binom{t}{n}$, and if you write this as a polynomial in t, you have to divide by $n!$. For simplices the coefficients will be positive but not necessarily integers. For some polytopes the coefficients may be negative.

There seems to be a contradiction — any polytope can be triangulated into simplices, and simplices have positive coefficients, but some polytopes cna have negatives. The reason is because you have to subtract for overlapping faces. You can write any polytope as a linear combination of simplices of various dimensions, possibly with negative signs; so the Ehrhart polynomial can have negative coefficients.

But there are some nice classes of polytopes whose Ehrhart polynomials always ar eintegers, or always have positive coefficients.

Question 13.15. What about Gelfand–Tstelin polytopes?

First, does Ehrhart theory apply to Gelfand–Tsetlin polytopes? The theorem requires that all vertices are integers — is it true that the GT polytope is a lattice polytope?

Let's try to figure out the vertices of $GT(\lambda)$. We know the top row is $\lambda_1, \ldots, \lambda_n$. Then each of the next entries has two conditions — μ_1 must satisfy $\lambda_1 \geq \mu_1 \geq \lambda_2$. For the vertices, you should degenerate all these inequalities to equalities. Of course you cannot require μ_1 to equal both λ_1 and λ_2; but for every entry, you have to pick one of the entries immediately above it. For example, our next row could be $\lambda_1, \lambda_2, \lambda_4, \ldots$. So every entry should be equal to one of the two entries above it. That means it is a lattice polytope, so Ehrhart theory applies. If λ_i are integers, then every entry on the vertices is equal to one of the λ_i.

Example 13.16

Find $L_{GT(n-1,n-2,\ldots,1)}$.

Solution. First, we can use the formula

$$\prod \frac{(j-i)(t+1)}{j-i} = (t+1)^{\binom{n}{2}}.$$

This means the number of lattice points is given by specializing $t = 1$.

Meanwhile, the volume is 1. What's another polytope with volume 1 and $2^{\binom{n}{2}}$ lattice points? Well, one is a cube.

What's the number of vertices? In our $n = 3$ answer, we can see that all but one of them are vertices — the one where one entry is not one of its parents is $210, 20, 1$. So the number of vertices is 7, which means this is not a cube — but it has the same volume, and the same number of lattice points.

We can actually draw a picture of

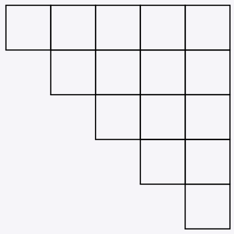
$$GT(2,1,0) = \{(x,y,z) \mid 2 \geq x \geq 1, 1 \geq y \geq 0, x \geq z \geq y\}.$$

Let's slightly change variables — write $\tilde{x} = x - 1$, and $\tilde{z} = z - y$, and $\tilde{y} = y$. Then our conditions are $\tilde{x}, \tilde{y}, \tilde{z} \geq 0$, $\tilde{x}, \tilde{y} \leq 1$, $\tilde{z} \leq \tilde{x} - \tilde{y} + 1$.

We can now draw a picture with this new picture. The base of the polytope will be a square in the $\tilde{x} - \tilde{y}$ plane. But then we have to lift each vertex, and z has certain conditions. For the origin we should have $z = 0$, forsome we should go one step up, and for some we should go 2 stpes up. So the result kind of looks like a cube, but one of the edges gets shrunk to a single point. □

In the remaining half of the lecture (it is 1:54), Prof. Postnikov was planning to show an application to something that looks unrelated. We will discuss this next time, but it is about the following objects, called *shifted* SYT:

Definition 13.17. A **shifted SYT** of shape $(n, n-1, \ldots, 1)$ is where we arrange the boxes as



where instead of left-justifying we guarantee that the left end is a staircase. We want to arrange numbers to be strictly increasing in rows and columns: for example, $123510, 46711, 8513, 1214, 15$. Now we look at the diagonal entries — here $(1, 4, 8, 12, 15)$. Can we characterize all possible diagonal vectors of shifted SYT?

THese actually form a nice polytope.

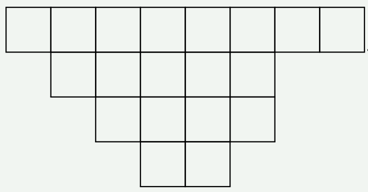
§14 October 7, 2022

Today we will discuss *shifted* shapes and tableaux.

Definition 14.1. Suppose $\mu = (\mu_1 > \mu_2 > \cdots > \mu_\ell)$ is a *strict* partition of $n = |\mu|$. Then the **shifted shape** is the same as a usual Young diagram, but we staircase-justify instead of left-justifying.

Example 14.2

The partition $\mu = (8, 5, 4, 2)$ is



Definition 14.3. A **SYT** of a shifted shape is filling the boxes of a shifted shape, without repetition, so that numbers are increasing in rows and columns. We use f^{sh}_μ to denote the number of SYTs of shifted shape $|mu$.

We can see that in our example, 1 must be in the top-left corner, and 2 must be next to it, but there's two choices for 3. For example:

$$\begin{array}{|c|c|c|c|} \hline 1 & 2 & 3 & 7 \\ \hline \multicolumn{1}{c|}{} & 4 & 5 & \multicolumn{1}{c}{} \\ \cline{2-3} \end{array}.$$

For normal SYT we had the hook-length formula. Interestingly, there's also a simple formula for shifted shapes, and it's very similar:

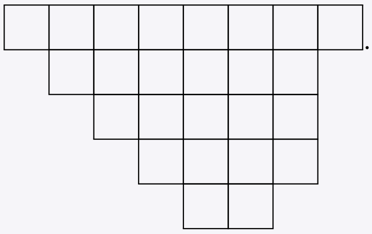
Theorem 14.4

$$f_\mu^{\text{sh}} = \frac{n!}{\prod_{x \in \mu} h(x)},$$

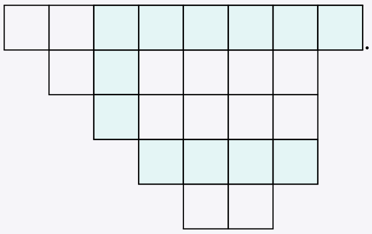
where $h*(x)$ is the hooklength of the box x.

This is practically the same formula, but we still need to define the hooklengths:

Definition 14.5. Suppose we have the shifted shape

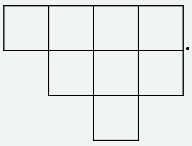


Then the *hook* is where we look at all boxes to the right or below, except that a hook may have a broken leg — if the hook hits the diagonal, then it bounces off it. (If the leg doesn't touch the diagonal, then it doesn't have a broken leg.)



Example 14.6

If $\mu = (4, 3, 1)$, then our shifted shape is



The hook lengths are

7	5	4	2
	4	3	1
		1	

The number of SYT is

$$f_\mu^{\text{sh}} = \frac{8!}{7 \cdot 5 \cdot 4 \cdot 2 \cdot 4 \cdot 3 \cdot 1 \cdot 1}.$$

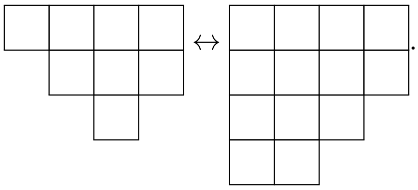
Again, there are many different ways to prove this. The same method as we used — using the volume of polytopes — still works.

In the usual hook-length formula, we found a bijection

$$\varphi_\lambda := \{\text{matrices of shape ydiagram}\} \leftrightarrow \{\text{reverse plane partitions of same shape}\}.$$

(The first means nonnegative, and the second means weakly decreasing and increasing. You can actually think of this as RSK.) We can use the same map, except that we need to specialize it to *symmetric* things: we assume $\lambda' = \lambda$ is symmetric with respect to reflection over the main diagonal, and then we can identify shifted shapes with usual self-conjugate shapes. We can restrict the RSK map to the cases when this matrix is symmetric and of course the RHS is also symmetric; then the same method will produce this broken leg hook length formula.

Note that in particular, shifted shapes correspond to usual self-conjugate shapes, by reflecting with respect to the main diagonal. For example,



Meanwhile, ifi we have a SYT of shifted shape

1	2	4
	3	5

then this corresponds to a tableau of a self-conjugate thing which isn't exactly standard — off-diagonal entries appear in pairs, like

1	2	4
2	3	5
4	5	

.

Note that there is a difference between how we treat entries on the diagonal and off the diagonal. So we can also count them in a more refined way, paying attention to the entries on the diagonal.

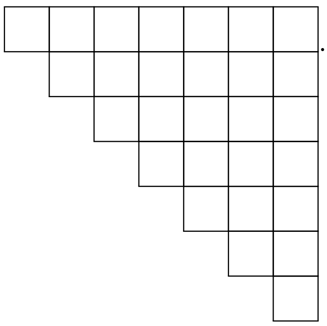
> **Question 14.7.** WHat can we say about the number of SYT of shifted shape with given (Fixed) entries $d_1, d_2, \ldots, d_\ell$ on the diagonal?

SO we have $\boxed{d_1}$ and we want to see how many ways there are to fill it to a SYT.

In the rest of the lecture, we will assume our shifted shape is the staircase shape. THe construction works in the general case, and many results can be generalized to arbitrary shifted shapes, but this is notationally simpler. So assume

$$\mu = (n, n-1, \ldots, 1)$$

is the **shifted staircases**



We want to fix $d_1, \ldots, d_1$ on the diagonal; we denote $N(d_1, \ldots, d_n)$ as the number of SYT of shifted shape $(n, n-1, \ldots)$ with the given diagonal entries.

First, a more basic question:

Question 14.8. When is this number nonzero — in other words, what are the possible diagonal entries?

Of course, there are some trivial observations: $d_1 = 1$, d_n is the maximal entry (which is $\binom{n+1}{2}$), and $d_1 < d_2 < \cdots < d_n$. Moreover, they're not just strictly increasing — if we have 1, then 2 has to be next to it, which means $d_2 \geq 3$. More generally, $d_{i+1} - d_i \geq 2$.

There are also other conditions. It will be more convenient to let $a_i = d_{i+1} - d_i - 1$, where $1 \leq i \leq n-1$ — these are the *differences* of diagonal entries, decreased by 1.

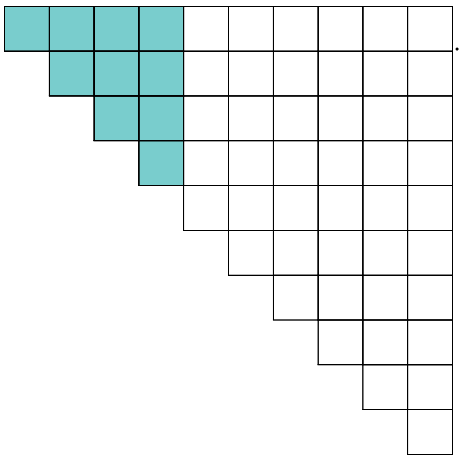
First let's look at one case where $N(d_1, \ldots)$ is easy to calculate. We know that $d_1 = 1$. Suppose we want to make d_2, d_3, … lexicograpically minimal. The minimal possibility for d_2 is 3; then we want d_3 as small as possible, which means we want it to be 6. This menas we get

1	2	4
	3	5
		6

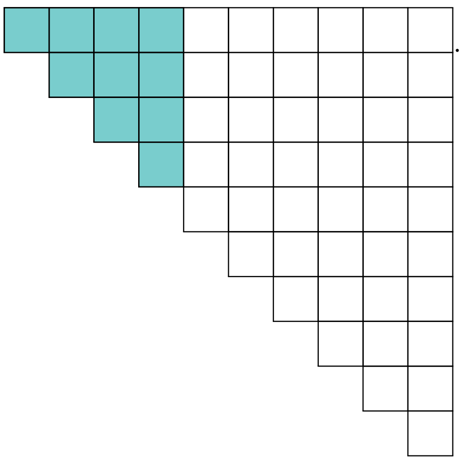
.

So for this vector $(1, 3, 6, \ldots)$ there's only one way to do it.

But now suppose we didn't want to minimize all of them — we wnated to only minimize the first k lexicographically, and maximize the rest. If we want to minimize the first k, that uniquely determines the top thing.



Then by maximizing the bottom diagonal, we fill in the bottom part:



So then it remains to fill in the remaining $k \times (n-k)$ rectangle.

We can do this by picking any usual SYT of this rectangular shape. So for this particular case, the numbers of shifted tableau actually become usual numbers of SYT of rectangular shape.

So then we let $a = (a_1, \ldots, a_{n-1})$ where $a_i \geq 1$ and $\sum a_i = \binom{n}{2}$. Then we write $\operatorname{diag}(a) = (d_1 = 1, d_2 = a_1 + 2, d_2 = a_1 + a_2 + 3, \ldots)$.

Proposition 14.9

$N(1, 2, 3, \ldots, k-1, M, n-k-1, \ldots, 3, 2, 1)$ where $M = k(n-k)$ is theh number of usual SYT for the rectangular shape $k \times (n-k)$, given by the usual hooklength formula.

Question 14.10. Are there other cases where this is explicitly given?

Now let's try to understand something more basic.

§14.1 Possible Diagonals

Now let's try to understand what the possible diagonals are.

Example 14.11

When $n = 3$, there are two shifted tableau:

1	2	4
	3	5
		6

and

1	2	3
	4	5
		6

.

So the possible diagonals are $(1, 3, 6)$ and $(1, 4, 6)$; and the a-vectors are $(1, 2)$ and $(2, 1)$.

Now let's characterize these possible diagonals in general.

First, some notation — we'll use polytopes, which all live in $\mathbb{R}^{n-1}$. We let $\Delta_{[i,j]}$ be the coordinate simplex given by $\operatorname{conv}(e_1, e_{i+1}, \ldots, e_j)$ (the convex hull of some of the coordinate vectors). Meanwhile, we define the polytope $\operatorname{Assoc}_{n-1}$ as the Minkowski sum

$$\sum_{1 \leq i < j \leq n-1} \Delta_{[i,j]}.$$

Definition 14.12. A **Minkowski sum** of two polytopes, and more generally two sets of points P and Q in $\mathbb{R}^{n-1}$, is $P + Q = \{p + q \mid p \in P, q \in Q\}$.

Remark 14.13. The name of this polytope is the **associahedron**; this is why it is called assoc. Or rather, there are many geometric realizations fo this thing, and this is one of them. You may know that the number of vertices of this polytope equals the Catalan numbers; you can try to figure this out.

Example 14.14

When $n = 4$, we want to find Assoc_3, which is a 2-dimensional polytope (it lives in the plane where coordinates sum to 1). This is the sum of the standard coordinate simplex — the triangle (convex hull of e_1, e_2, e_3), and then we have $\Delta_{1,2}$ which is the line segment e_1e_2, and $\Delta_{2,3}$. So we have a equilateral triangle (pointing up), and then we have its left and right edge. We also have the single points $\Delta_{[1,1]}$ and so on, which together shift the thing by $(1, 1, 1)$.

To calculate this, we start with our triangle. Then we want to add any point on this interval to this triangle. So we shift our triangle in the direction of this interval. Then we shift what we get here in the direction of that other vector.

We can explicitly write down coordinates: $(3, 2, 1)$, $(3, 1, 2)$, $(2, 1, 3)$, $(1, 2, 3)$, $(1, 3, 2)$, $(2, 3, 1)$, and in the middle we have $(2, 2, 2)$, and at the top we have $(1, 4, 1)$. So we have eight lattice points.

Theorem 14.15

These vectors are exactly all possible vectors for which N is nonzero. More explicitly,

$$N(\text{diag}(a_1, \ldots, a_{n-1})) \neq 0$$

if and only if $(a_1, \ldots, a_{n-1})$ is a lattice point of the associahedron — if it's in $\text{Assoc}_{n-1} \cup \mathbb{Z}^{k-1}$.

Note that if you remove the top vertex, in this case you get the permutahedron.

Now, how do we prove the theorem, and how do we count these numbers — what's the relationship between this and these other things?

We'll now return to stuff we discussed in the previous lecture — namely, **Gelfand-Tsetlin polytopes**. We'll look at $GT(\lambda)$, which lives in $\mathbb{R}^{\binom{n}{2}}$ and is the polytope consisting of all points whose entries satisfy all the linear inequalities used to deifne a GT pattern. Note that this λ is not related to λ we have elsewhere in this lecture. It does not matter whether these λ_i are integers or not; they can be arbitrary real numbers (so we are not going to treat these as a partition, but rather as parameters of a GT pattern).

Last time, we saw that $\text{Vol}(GT(\lambda))$ is given by the simple multiplicative formula

$$\prod_{1 \le i < j \le n} \frac{\lambda_i - \lambda_j}{j - i}$$

(as given by Weyl's formula). (Note that the λ_i are weakly decreasing.) This formula is true not just for integers but for arbitrary real numbers; so we will now assume our λ's are generic, meaning they are all distinct.

Now let's count Vol in a different way, by subdividing it into simple pieces. These simple pieces would be given by all possible total orderings of the entries in a GT pattern. Let's again assume it's a generic thing; all entries are different, but come in a certain order.

If you think about it, all possible linear orderings of entries are exactly teh same thing as shifted SYT. (We have to rotate and reflect a bit, but this is easy to see.)

So possible linear orders of the entries in a GT pattern are in bijection with shifted SYT's of this triangular staircase shape $(n, n-1, \ldots, 1)$. For example, the standard young tableaux

1	2	4
	3	5
		6

means we have then 1 corresponds tot he rightmost dot, 2 to the and so on — so this corresponds to

$$\begin{array}{ccc} x+6n & x_3 & x_1 \\ x_5 & x_2 & x_4 \end{array}.$$

where $x_1 < x_2 < x_3 < x_4 < x_5 < x_6$. We also know that $x_1 = \lambda_3$, $x_3 = \lambda_2$, and $x_6 = \lambda_1$ in this example. So basically we have an increasing sequence of numbers, and some of them are fixed — the positions which are fixed exactly correspond to entries of the shifted tableaux on the diagonal.

So nwo let's see what our polytope looks like: $X_1 < x_2 < x_3 < x_4 < x_5 < x_6$ where x_1, x_3, x_6 are fixed. This thing will be a product of simplicies: first we have a one-dimensional simplex, the line segment between λ_2 and λ_3 — that's a 1-dimensional simplex Δ^1 dilated by $\lambda_2 - \lambda_3$. And then we have a two-dimensional simplex from $\lambda_2 < x_4 < x_5 < x_6$. So we get

$$(\lambda_2 - \lambda_3)\Delta^1 \times (\lambda_1 - \lambda_2)\Delta^2.$$

Every shifted SYT corresponds to one piece of a GT pattern, and these things are products of dilated simplices. (In this example, we have a triangular prism.)

Last lecture, for $n = 3$ we calculated what $GT(\lambda)$ is — it looks like a sort of cube where we have a square base and then we have a slanted top thing. (Toby said that there is a building like this in Chicago, so we now call it the Chicago polytope.)

This polytope is now subdivided into two pieces that look like triangular prisms. Figuring out how to subdivide it is left as an exercise.

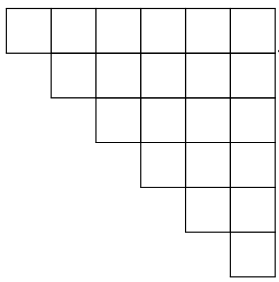
The rest is that we can subdivide GT, an therefore write this product as a sum over all shifted tableau of products of simplices, and for each product of simplices the product is given by a simple expression. We can compare these for the same thing, adn we will deduce this theorem and actually an explicit formula for these numbers in many cases.

§15 October 12, 2022

Last time, we discussed $\mathrm{GT}(\lambda)$, the set of all real-valued GT patterns with top row λ, which is a polytope in $\mathbb{R}^{\binom{n}{2}}$. We saw that there's two ways to write down its volume. One is

$$\mathrm{Vol}(GT(\lambda)) = \prod_{1\le i<j\le n} \frac{\lambda_i - \lambda_j}{j-i} = \frac{1}{1!\,2!\cdots(n-1)!}\prod_{i<j}(\lambda_i - \lambda_j),$$

using the Weyl dimension formula. Another is that we can rotate the GT pattern and transform it into a shape that looks like a SYT of shifted shape:



For each shape, the diagonal entries $d_1 \le d_2 \le \cdots$ are special. So we can write this as

$$\sum_{d_1,\dots,d_n} \sum_{T\text{SYT of shifted shape with diagonal } d_1,\,\dots,\,d_n} \mathrm{Vol}((\lambda_{n-1} - \lambda_n)\Delta^{d_2-d_1-1} \times (\lambda_{n-2} - \lambda_{n-1}) \times \Delta^{d_3-d_2-1} \times \cdots).$$

since each piece is a product of simplices. This is easy to calculate — the volume of a simplex is easy to calculate, and the volume of a product is the product of volumes. So we can write this as

$$\sum_{d_1,\ldots,d_n}\sum_T\prod_{i=1}^{n-1}\frac{(\lambda_{n-i}-\lambda_{n-i+1})^{d_{i+1}-d_i-1}}{(d_{i+1}-d_i-1)!}.$$

For example, fo r$n=3$ we saw that the GT polytope looks like the Chicago skyscraper shape (square bottom, tilted top). We can do this by taking the leftmost triangle and pushing it to the right; that gives us one piece, and the rest is also a triangular prism.

Comparing these formulas, we can write down teh followign theorem:

first, here we have differences of λ and of d, so we will change variables — write $a_i = d_{i+1}-d_i-1$, and $t_i = \lambda_{n-i}-\lambda_{n-i-1}$ (for $1\le i\le n-1$). Rewriting this equality in these variables gives:

Theorem 15.1

We have

$$\sum_{a_1,\ldots,a_{n-1}\ge 1,\sum a_i=\binom{n}{2}} N(\mathrm{diag}(a_1,\ldots,a_{n-1}))\frac{t_1^{a_1}}{a_1!}\cdots\frac{t_{n-1}^{a_{n-1}}}{(a_{n-1}!)} = \frac{1}{1!\,2!\cdots(n-1)!}\cdot\prod_{i<j}(t_i+t_{i+1}+\cdots+t_{j-1}).$$

where $\mathrm{diag}(a_1,\ldots,a_{n-1})$ is the corresponding sequence of d's (so $(1, a_1+2, a_1+a_2+3, \ldots)$) and $N(d_1,\ldots,d_n)$ is the number of shifted tableau with the given diagonal entries.

The point is that the important thing is this product — up to some explicit multiplicative factor, these numbers of shifted SYT with given diagonal entries are basically coeffiients in the expansion of this product.

Definition 15.2. GIven a polynomial $f(t_1,\ldots,t_m)=\sum a_{i_1,\ldots,i_m}t_1^{i_1}\cdots t_m^{i_m}$ (here $m=n-1$), its **newton polytope** is teh convex hull of all integer vectors $(i_1,\ldots,i_m)$ such that the coefficient $a_{i_1,\ldots,i_m}$ is nonzero.

Example 15.3

The Newton polytope of $1+25x+70xy^2+\sqrt{\pi}x^3$ is the convex hull of the points $(0,0)$, $(1,0)$, $(1,2)$, and $(3,0)$. We can draw this out, and we get a triangle.

Many asymptotic features of the polynomial (how fast it grows if coordinates increase in a direction) are governed by this polytope — ew can think of it as a multidimensional generalization of degree.

Definition 15.4. A polynomial f has the **saturated Newton polytope (SNP) property** if for all lattice points of the Newton polytope, t he corresponding monomial coefficient in f is nonzero. In other words For any $(i_1,\ldots,i_m)\in\mathrm{Newton}(f)\cup\mathbb{Z}^m$ ew have $a_{i_1,\ldots,i_m}\neq 0$.

Example 15.5

The above example does not have the SNP property because we haven't marked all lattice points of the Newton polytope — we are missing $(1,1)$, $(2,0)$. But if we added extra monomials then it would have the property.

We would not expect a generic polynomial to have this property, but some polynomials we discuss in this class do.

Theorem 15.6

The Schur polynomial has the SNP property.

Recall that the Newton polytope of the Schur polynomial is the permutahedron. SO for any lattice point in the permutahedron, the coefficient (the Kostke number) is nonzero. The first problem on the problem set is basically this theorme, phrased in terms of Kostke numbers — the Kostke number $K_{\lambda,\mu}$ is nonzero iff $\mu \le \lambda$ in the dominance order.

There is another generalization of this, that is an open problem — for example, epople looked at Schubert polynomials. For some classes they proved the SNP property; Prof. Postnikov thinks it's still an open problem to prove that any Schubert polynomial has the SNP property.

A problem that may be added to the problem set:

Lemma 15.7

This product

$$\prod_{1\le i<j\le n} (t_i + t_{i+1} + \cdots + t_{j-1})$$

has the SNP property.

To see what the Newton polytope of this thing is, note that if we take the Newton polytope of the product of two polynomials, we take the Minkowski sum —

$$\text{Newton}(fg) = \text{Newton}(f) + \text{Newton}(g).$$

This makes sense because Newton is a generalization of degree, and the degree of products is the sum of degrees. Here we have a product of many linear terms, and for each the Newton polynomial is a coordinate simplex; this means the Newton polytope of this guy is

$$\sum_{i<j} \Delta_{[i,j-1]}$$

which we last time called the **associahedron**. There are many ways to realize teh associahedron and this is one of them.

If we believe this lemma, then we have a characterizatino of all diagonal entries of all shifted tableaux: these correspond exactly to all lattice points of the associahedron.

Moreover, we also in many cases can calculate explicitly this number — up to some factor (the product of factorials), it's the coefficient in the expansion.

Corollary 15.8

$N(\text{diag}(a_1, \dots, a_{n-1})) \ne 0$ if and only if $(a_1, \dots, a_n) \in \text{Assoc}_{n-1} \cap \mathbb{Z}^{n-1}$.

Corollary 15.9

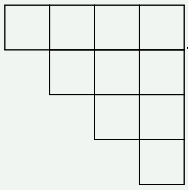
$$N(\text{diag}(a_1, \dots, a_{n-1})) = \frac{a_1! \cdots a_{n-1}!}{1! \cdots (n-1)!} \cdot \text{some integer factor},$$

where this integer is the coefficient of the corresponding monomial in the expansion of this product.

We can actulaly say more about this thing. For any *vertex* in the Newton polytope, its coefficient is 1 — the Minkowski sum of several things will be some thing, and for every vertex there's a unique way to write it down. In this case, it will be if and only if — for every non-vertex the coefficient will be at least 2. So then this integer factor equals 1 if and only if $(a_1, \dots, a_{n-1})$ is a vertex of hte associahedron. It is another exercise to show that the associahedron has the Catalan number of vertices.

Example 15.10

When $n = 4$, we want to count the number of shifted SYT of shape



.

Suppose we have diagonal $(1, 3, 8, 10)$. The corresponding a-vector is $(1, 4, 1)$. If we have 3 here then 2 should be above, if we have 8 then 9 should be there, and in the 2×2 we can pick any SYT of 2×2 shape. There are two such things, so then the number is $N(1, 3, 8, 10) = 2$. Meanwhile, the expression tells us

$$\frac{1!\,4!\,1!}{11!\,2!\,3!} \cdot \text{some integer}$$

and here that integer factor is 1, since this is a vertex. We can see this is 2.

Last time, we mentioned that more generally if we pick the first k diagonal entries as small as possible that fixes the top-left triangle, and if we require that the rest are as big as possible that fixes the bottom triangle; t hen the number of ways to fill in the shape is the number of SYT of shape $k \times (n - k)$. This actually happens to be a vertex, so in this case the coefficient is 1 and we get an explicit multiplicative formula for the number of SYT of rectangular shape. This is in general given by the hook-length formula. We already proved the hook-length formula, but here we got it without using any nontrivial stuff, at least for rectangular shapes.

So in some special case, this specializes to the usual hook length formula for the rectangular shape.

If we rewrite this product in the old coordinates $\lambda_1, \dots,$ then that product becomes $\lambda_i - \lambda_j$. So then we get the Vandermonde determinant, whose lattice points correspond to permutations. So the permutahedron is the newton polytope of this polynomial in the λ_i, and the associahedron in the t_i.

§15.1 Young's Lattice

Definition 15.11. Young's Lattice is the poset of all Young diagrams ordered by inclusion.

It has a single minimal element $\emptyset$, then one thing with a single box, then two things with two boxes, and so on. We denote it by $\mathbb{Y}$.

Note that we called this a **lattice** — that's a special kind of poset which has meet and join (for any two elements, there's a unique minimal element above them, which is their *join*, and a unique maximal element below both, called *meet*). It's easy to see that Yougn's lattice is a lattice, because meet and join are just the set-theoretic union and intersection of two Young diagrams.

Definition 15.12. A covering relation in $\mathbb{Y}$ is $\lambda \lessdot \mu$ iff $\mu \supset \lambda$ and their difference is just a single box.

We can define two operators, the up and down operators U and D, acting on the space $\mathbb{C}[\mathbb{Y}]$:

Definition 15.13. $\mathbb{C}[\mathbb{Y}]$ is the space of formal linear combinations of Young diagrams.

For example, elemnts of htis may be

$$\sqrt{\pi}\cdot \ydiagram{1} - 3\cdot \ydiagram{2,1} + 100\varnothing.$$

The most confusing thing is that the empty shape is not 0.

Definition 15.14. U sends $\lambda \mapsto \sum_{\mu \gtrdot \lambda} \mu$ (the sum of all things obtained by adding one box), and D sends $\lambda \mapsto \sum_{\mu \lessdot \lambda} \mu$ (the sum of all things obtained by deleting a box).

These two operators satisfy a very nice relation:

Lemma 15.15

The commutator $[D, U]$ is the identity operator $\mathbf{1}$.

By definition, the **commutator** is $[D, U] = DU - UD$. So these two things don't commute, but they almost commute.

Proof. Suppose we want to calculate $[D, U](\lambda) = DU(\lambda) - UD(\lambda)$. The first term counts the number of all possible ways to add a box to λ and then remove a box from λ; the second first removes a box and then adds a box.

So we start with λ and get to ν. In the first thing we go up $\lambda \to \mu$ and then down $\mu \to \nu$; in the second we go down first to $\tilde{\mu}$ and then up to ν. The claim is that the number of such μ's is the number of such $\tilde{\mu}$'s when $\lambda \neq \nu$ — nonzero things can only happen on the diagonal. So we basically want to show two conditions:

- If $\nu \neq \lambda$, if there are k ways to go up and down from λ to ν, then there's also k ways sto go down and up. This is slightly misleading becaause the number is always 0 or 1.
- If $\lambda = \nu$ (so we go up and then down to teh same Young diagram), then the difference equals 1.

So in other words, we should have $k+1$ μ's above λ, and k $\tilde{\mu}$'s below it. SO the number of stuff which cover λ is one more than the number of stuff covered b yλ.

In Young's lattice, it's easy to see that both properties are true. In the first case, we add some box, and then we remove some other box:

$$\ydiagram{5,3,2,1}$$

Since these boxes are different boxes, we could reverse the order — we could instead remove and then add.

On the other hand, property 2 says if we want to add a box and then remove the same box, the number of ways to do this is the number of inner corners of λ. Meanwhile the stuff we can remove are the outer corners. We can easily see that in a Young diagram, the number of outer corners plus one is the number of inner corners, which is basically what this relation says. $\square$

As we will see, this relation implies many nice properties. There are also more general operators and relatison — add and remove horizontal and vertical strips. In the problem set there is a problem about these more general operators, that should be generalizations of this.

§16 October 14, 2022

§16.1 Differential Posets

Definition 16.1. A **differential poset** P is a partially ordered set such that:

- P is a locally finite ranked poset — there is a rank function $\mathrm{rk}\colon P \to \mathbb{Z}_{\geq 0}$, such that any edge of the Hasse diagram goes between two adjacent ranks. In other words, whenever $\lambda \lessdot \mu$, $\mathrm{rk}(\mu) = \mathrm{rk}(\lambda) + 1$. (Locally finite means that P has finitely many elemnts of rank n for any given n.)
- P has a unique minimal element $\hat{0}$, with $\mathrm{rk}(\hat{0}) = 0$.
- With the up and down operators acting on $\mathbb{C}[P]$ (the space of formal linear combinations of elements of P — we could use any field instead of P) such that $U\colon \lambda \mapsto \sum_{\mu \gtrdot \lambda} \mu$ and $D\colon \lambda \mapsto \sum_{\mu \lessdot \lambda} \mu$, these operators satisfy the relation
$$[D, U] = I.$$

The last condition — that $[D, U] = I$ — is the important one.

The reason for 'locally finite' is so that eveyr elemnet has finitely many elements above and below it, so that U and D make sense.

Last class, we saw the following:

Lemma 16.2

Young's lattice is a differential poset.

We saw earlier that the relation $[D, U] = I$ means two things:

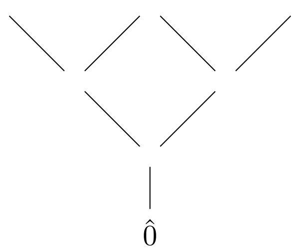
1. For all $\lambda \neq \mu$ in P, the number of elements which cover both λ adn μ is equal to the number of elements which are covered by both. (We can draw λ and μ next to each other, and k dots above and below, with lines.) In $\mathbb{Y}$ k is at most 1.
2. When $\lambda = \mu$, the number of elements that cover λ is 1 more than the number of element socvered by λ. So if there are k things below it (again as dots), there should be $k + 1$ things above.

Student Question. HOw does D act on $\hat{0}$?

It sends it to 0 (we should not confuse $\hat{0}$ with 0), since there is nothing below it.

In fact, we can start by writing down $\hat{0}$. Then there's nothing below it, so there should be one thing above it. Then there's one thing below that, so there should be 2 above it. So the first two levels are unique.

The next is also — there is a way to go down and up, so there should be exactly one way to go up and down. Then for each of these there is one thing below it, so there should be two above it; so we should add two extra things.



Exercise 16.3. Are there differential posets other than $\mathbb{Y}$?

A lot of nice things about $\mathbb{Y}$ are true for any differential posets.

Definition 16.4. For any differential poset, f_λ^P is the number of saturated chains $\hat{0} <\!\cdot\, \lambda^{(1)} <\!\cdot \cdots <\!\cdot\, \lambda^{(n)} = \lambda$.

In the case of Young's lattice, this is the number of SYT.

Theorem 16.5

We have

$$\sum_{\mathrm{rk}(\lambda)=n} (f_\lambda^P)^2 = n!\,.$$

We saw how to prove this for $\mathbb{Y}$ using Robinson–Schensted correspondence, but there's a fiarly simple proof for *any* differnetial poset.

Proof. Suppose we have two paths from $\hat{0}$ to λ. Then we can reverse the second — so we can think of pairs of paths from $\hat{0}$ to λ as the number of loops starting at $\hat{0}$, going n steps up to λ, and then going n steps down back to $\hat{0}$. This means the left-hand side is the coefficinet of $\hat{0}$ in $D^n U^n(\hat{0})$.

These up and down operators should satisfy the relation $DU = UD + 1$. There is another obvious condition, that $D(\hat{0}) = 0$.

Now we claim that these two properties formaly imply that this expression equals $n!\,\hat{0}$. To see this, we can kind of write this as

$$\underbrace{DDD\cdots D}_{n}\underbrace{UUU\cdots U}_{n}(\hat{0}).$$

We can kind of thin of these up adn down operators as particles hopping over each other. Take the rightmost D; we'd like to move it all the way to the right. At each step it can teitehr jump over U (corresponding to UD) or it can annihiliate it (corresponding to the 1 — they bump into each other and explode and kind of disappear, as if the up operator is a particle and the down operatora n antiparticle).

So the first D may jump over the first U, or over the second, and so on. But if it jumps lal the way to the right, then we get 0, since $D(\hat{0}) = 0$. So the D has to annihilate with some U.

So then the thing we're trying to find is the number o fmatchings between all U's and all D's, times $\hat{0}$. There are n U's and n D's, so there's $n!$ ways to match them. ☐

There's an even shorter way to prove this identity, if you are not convinced by U's jumping over D's; this also explains why differential posets are claled differential (they should have some relation to differentiation).

Consider the two operators acting on the space of polynomials in one variable $\mathbb{C}[x]$. Let X be the operator $f(x) \mapsto xf(x)$, and D be the operator $f(x) \mapsto f'(x)$. Then we can easily see that X and D satisfy exactly the relation $[D, X] = 1$, and $D(1) = 0$. So then X and D satisfy exactly the same relations as the up and down operators for a differential poset; the only difference is in this model, hte unique minimal element is 1 (so $\hat{0} = 1$, D corresponds to D, and U to X).

But as we just saw, the two relations formally imply what the coefficient is. SO in stead of using the up and down operators, we can use these operators on the polynomial. Then we're trying to find

$$D^n X^n 1 = D^n x^n.$$

So we take the nth derivative of x^n n times, and this gives $n!$.

Student Question. Why is it clear that all up and down operators will give the same result?

We can start with $DD\cdots DUU\cdots U(\hat{0})$ and try to move all the D's to the right. Using the two relations, we can see that we'll end up getting some fixed coefficinet times $\hat{0}$. So any two operators satisfyign teh same relation will produce the same coefficient.

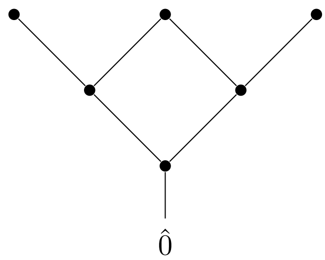
Student Question. Can you get a bijection between paths?

If you fix some kind of bijective proof of $DU = UD + 1$ — for any λ and μ you make a bijection between ways to go up and down, adn ways to go down and up (the k elements on the top and bottom for λ and μ, and the k wlements on the bototm and k of the stuff ont he top), then yo ucoud apply this and get the Schensted correspondence for your differential poset. (There are many possible bijections but if you make a chioce of correspondence between your stuff, then that uniquely defines a bijection — this is another exercise.)

§16.2 Building a Differential Poset

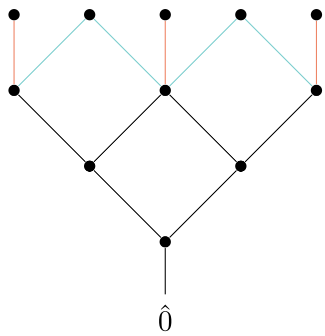
Question 16.6. Is there only one differential poset?

Previously, we had this thing:



The natural thing to do would now be to flip our thing over, and then add one extra element above each thing.

So we take the $(n-1)$th level and reflect it over the nth level (this makes condition 1 hold), and then we add one extra thing above all our vertices (this makes condition 2 hold).



As we can see, we have 1, 1, 2, 3, 5, 8, 13 elements. We probably know this sequence of numbers; these are called the **Fibonacci numbers**, where $F_0 = F_1 = 1$ and $F_n = F_{n-1} + F_{n-2}$.

n	0	1	2	3	4	5	6	7
$p(n)$	1	1	2	3	5	8	13	21

Meanwhile, the number of ele,ents of rank n in Young's lattice is the number of partitions of size n. These are called the **partition numbers** $p(n)$.

n	0	1	2	3	4	5	6	7
$p(n)$	1	1	2	3	5	7	11	15

So these things start as the Fibonacci numbers, but then they become smaller than the Fibonacci numbers.

So our construction is *not* unique — when constructing a differential poset, there *are* some choices we can make.

Definition 16.7. The poset we constructed is called the **Fibonacci lattice** and denoted by $\mathbb{F}$.

As we have seen, $\mathbb{F} \neq \mathbb{Y}$, because their rank numbers are different.

Exercise 16.8. Show that $\mathbb{F}$ is a lattice.

When we did our construction, we always assumed that this part of the poset, between the $(n-1)$th and nth level, should be reflected. But actually this condition doesn't imply it should be reflected — just that the number of ways to go up and down should equal the number of ways to go down and up. That doesn't automatically imply the two parts should be the same.

§16.3 Oscillating Tableaux

Instead of just looking at UUUDDD, we can take any sequence. This gives *standard oscillating tableaux*.

An oscillating tableaux is an arbitrary path in Young's lattice, or any differential poset — we start at some partition adn add a box or remove a box.

For example, let's suppose we fix a **walk** $W = W_{2n}W_{2n-1}\cdots W_2W_1$, such that $W_i \in \{U, D\}$ for each letter. Let's say we have n U's and n D's.

Example 16.9

We can take the walk $W = DDDUUDUU$.

Question 16.10. What is the number of paths in the Hasse diagram of $\mathbb{Y}$ (or any differential poset) that start and end at $\hat{0}$ and have the prescribed sequence of up and down steps — up and down steps given by the word W?

The same argument shows that this number of oscillating tableaux — starting and ending at 0 and going up and down in a certain way — equals the coefficient of $\emptyset$ (the empty partition) in $W(\emptyset)$ (where we think of W as the product of its up and down operators). What that means combinatorially is, using the same argument with U's and D's hopping over U, the number of matchings between all U's and all D's, such that every D is matched with some U to its right (because D's can only annihilate with U's to the right of themselves, since they hop to the right).

Example 16.11

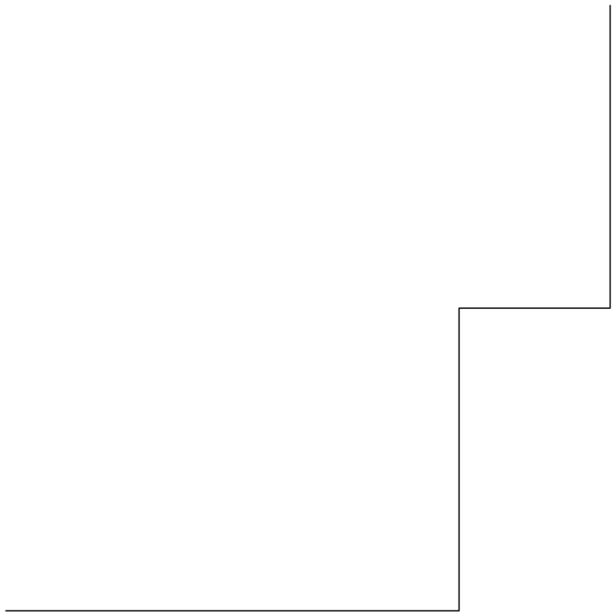
Take the sequence $DDDUUDUU$.

For convenience, index this as $D_1D_2D_3U_1U_2D_4U_3U_4$.

Then each D can be matched with some U to the right of it: you can draw these by a bunch of arcs.

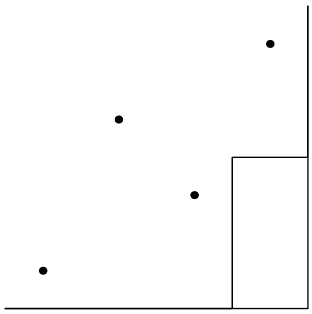
We can convert this matching into another thing: we can construct a Young diagram given by this sequence of up and down steps.

Start at some point. Whenever we see D, go one step to the right; whenever we see U, go one step up. This gives us RRRUURUU:



Call this thing κ_W (it's called κ and determiend by the word). Then whenever we see an edge between D and U, we place a rook in that box.

So if D_1 is matched to U_1, D_2 to U_3, D_3 to U_2, and D_4 to U_4, then we get:



So then this number we were thinking about is the number of placements of n non-attacking rooks on the Young diagram κ_W.

> **Example 16.12**
> What is the number of paths in $\mathbb{Y}$ $\varnothing = \lambda^{(0)}, \lambda^{(1)}, \dots, \lambda^{(2n)} = \varnothing$, such that for all i, at each odd step we either stay at the same point or go one step up ($\lambda^{(2i+1)} = \lambda^{(2i)}$ or $\lambda^{(2i+1)} = \lambda^{(2i)} \cup \text{box}$), and at even steps, $\lambda^{(2i)}$ equals either $\lambda^{(2i-1)}$ minus a box or $\lambda^{(2i-1)}$. So at every step we can either stay at the same position, or go one step up or down depending on the parity of I. We want to return back to the empty partition.

So in other words, we are applying

$$\cdots (U+1)(D+1)(U+1)\emptyset$$

where there are $2n$ factors. If we apply this sequence we don't necessarily come back to the 0th level, but this is possilbe, so then this is some coefficient times $\emptyset$ plus somethign else.

This is actually a famous combinatorial sequence which has not appeared in this class yet.

§17 October 17, 2022

§17.1 r-Differential Posets

A r-differential poset is a generalization of a differential poset. The definition is the same, except that $[D, U] = rI$ instead of just I.

Many things we said about differential posets can be generalized to r-differential posets in a straightforward way, with an extra factor of r.

Theorem 17.1
We have

$$\sum_{\lambda \in P, \mathrm{rk}(\lambda)=n} (\#\text{saturated chains in } P \text{ from } \hat{0} \text{ to } \lambda)^2 = r^n \cdot n!\,.$$

The proof is basically the same.

Now let's see some examples of r-differential posets.

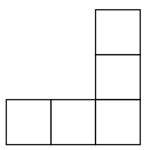
Definition 17.2. $\mathbb{Y}^{(r)}$ is the set of all Young diagrams, where the covering relation is that $\lambda \lessdot \mu$ if λ is a subdiagram of μ, and $\lambda \setminus \mu$ is a r-ribbon.

So we add not just a single box, but r boxes that form a ribbon.

Example 17.3
For $r = 2$, we add either a vertical or horizontal domino.

Note that for $r > 1$, this poset is not connected (and therefore it cannot be a differential poset, because it does not have a unique minimal element).

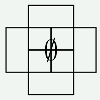
By definition, a ribbon is a connected path thing, such as



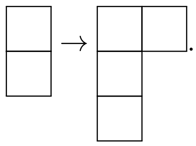
Lemma 17.4
Each connected component of this poset is a r-differential poset.

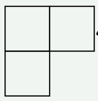
Example 17.5

For $r = 2$ we have:



We can see that for example, for the horizontal domino on the right, there's one edge going up and there's three going up. For example not ethat there is no edge



Student Question. Is there anything below ?

No — it is impossible to remove a domino from this partition. This poset actually has many minimal elements — it belongs to a different connected component. But we can still start adding dominos to this guy — and if we continue this we notice that the graph we would get is actually isomorphic to this one. There's another minimal element — a single box — and we could also add horizontla and vertical dominoes to the single box, and we'd get another connected component.

As we can notice, there are infinitely many minimal elements — any staircase Young diagram is a minimal element, since it's impossible to remove a domino.

Student Question. How do you know those various odd ones don't meet higher up — i.e. how do you know there isn't some relaly big diagram bigger than both the single box and the staircase?

Proposition 17.6

Show that each connected component of $\mathbb{Y}^{(2)}$ has a unique minimal element, which is a staircase, and each connected component is isomorphic to the product of $\mathbb{Y} \times \mathbb{Y}$.

If you want, you can solve this on the pset. Moreover, a similar claim is true for any r:

Proposition 17.7

Each connected component has a unique minimal elemnt, and is isomorphic to the product of r copies of $\mathbb{Y}$.

These minimal elements have a special name.

Definition 17.8. The minimal elements of $\mathbb{Y}^{(r)}$ are called r-cores.

For example 2-cores are staircases.

Lemma 17.9

The following are equivalent: λ is r-core iff λ has no hook-length $h(a) = r$, and equivalently if λ has no hook-length *divisible* by r.

In particular if you have a hook length that is *divisible* by r, then you have a hook length that's *equal* to r. The equivalence between (1) and (2) is kind of obvious — r-cores are partitions whose Young diagrams we cannot remove a r-ribbon, and recall that these removable ribbons have length exactly the hook lengths. (We saw this when proving the hook length formjula.)

Remark 17.10. At least three problems on the pset are related to this stuff for $r = 2$. We prove this formula for standard domino tableaux; one way suggested is the operators of adding and removing horizontal strips. There is a more straightforward way if you use these 2-differential posets, but then you have to prove some fo tehse lemmas. Another related problem is to describe all elements that belong to the connected component of $\emptyset$ — all partitions that can be obtained fro $\emptyset$ ny adding dominos. There is another problem about inventing Schensted correspondence for domino tableaux.

Proposition 17.11

For any r-differential poset P,

$$\#\text{ chains } \lambda^{(0)} = \emptyset, \lambda^{(1)}, \dots, \lambda^{(2n)} = \hat{0}$$

such that at every step we go either up or down, meaning we add a domino or remove a domino (or more generally a r-ribbon) (such that $\lambda^{(i+1)} \cdot\!> \text{ or } <\!\cdot\, \lambda^{i}$). The number of such chains equals

$$r^n(2n-1)!!.$$

Reclal that $(2n-1)!! = 1 \cdot 3 \cdot 5 \cdots (2n-1)$. We may know that this has a nice combinatorial itnerpretation — it si the number of perfect mathcings in K_{2n}. (YOu have $2n$ points 1, 2, ..., $2n$, and you wnat ot match them in pairs by drawing a bunch of arcs.)

SO we are talking about the coefficinent of $\hat{0}$ in $(U+D)^{2n}(\hat{0})$. In the next example:

Proposition 17.12

For any r-differential poset P,

$$\#\text{chains } \lambda^{(0)} = \hat{0}, \dots, \lambda^{(2n)} = \hat{0}$$

such that every odd step either covers or is equal to the previous thing, and every even step either is covered by or is equal to the previous thing. (This is the example from before.)

This number equals

$$\sum_{\text{set partitions } \pi \text{ of } [n+1]} r^{n+1-\#\text{blocks of } \pi}.$$

In particular if $r = 1$ this is the **Bell number** $B(n+1)$, defined as the number of set partitions of $n+1$.

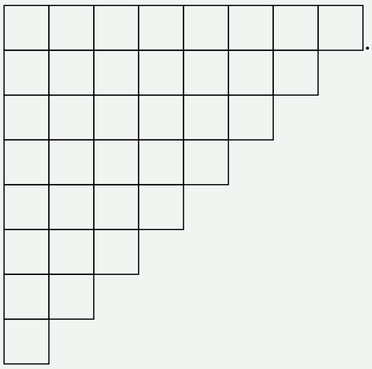
In terms of up and down operators, we are looking at the coefficient of $\hat{0}$ in $((D+1)(U+1))^n\hat{0}$.

For a general r this is the value of the Bell polynomial — this will be $r^{n+1}B_{n+1}(r^{-1})$.

We can apply the techique from last lecture — on one hand these correspond to matchings, and to rook placements. So we are talking about rook placements where up-steps correspond to rows and down-steps to columsn. Then we end up getting a rook placement on the triangular shape $(n, n-1, \dots, 1)$.

Example 17.13

We have a staircase shape



We are no longer required to ahve a rook in each row adn column — we just need to place any number of non-attacking rooks on this shape. (If we wanted to place the maximum, they'da ll have to go on teh diagonal which is fairly easy.)

On the other hand, we claim these things are in bijection with set partitions of $[n+1]$. Remember that a rook corresponds to the case where the U kills the D, so placing a rook gives you a factor of r. SO r counts these rook palcements with weight $r^{\text{number of rooks}}$.

Now to go from a rook placement to a set partition, we write numbers in the corners: 9, 8, ..., 1 on the main diagonal. Then we place rooks, and at each rook we place a hook, drawing one leg and one arm (down and right).

Example 17.14

WE could have a rook at 1–3, 3–7, 4–8, 8–9. This corresponds to $\pi = (137 \mid 489 \mid 2 \mid 5 \mid 6)$.

This construction gives a one-to-one correspondence between rook placements with an arbitrary numer of rooks, and set partitions of $n+1$. We can see that the number of rooks under thsi correspondence is $n+1$ minus the number of blocks in π. So this gives you r to the number of rooks, which is r to $n+1$ minus the number of blocks.

Remark 17.15. After the pset we will probably spend one lecture on presentations; any volunteers can present a solution to a problem. This will be next week.

Remark 17.16. There is a problem about deriving the Weyl character formula; some of us should do this problem and present it.

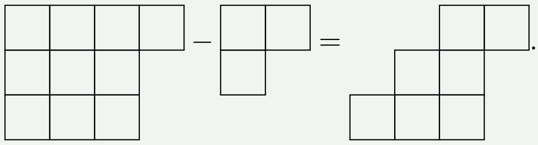
§17.2 Jacobi–Trudi Formulas

Earlier we saw how to express elementary and homogeneous symmetric polynomials in terms of each other. This can be done recursively, but it can also be done in terms of determinants. This can actually be done more generally — we can express any schur function in terms of elementary and complete homogeneous functions, and even for *skew* sf's.

Definition 17.17. Suppose we have two partitions λ and μ such that $\lambda \supseteq \mu$. THen we define the **skew Young diagram** $\lambda \setminus \mu$ as the set-theoretic difference of the set of boxes of λ and μ.

Example 17.18

If $\lambda = (4,3,3)$ and $\mu = (2,1,0)$ then we get



FOr skew shapes, we can define skew schur functions in exactly the same way (the combinatorial definition):

Definition 17.19. $s_{\lambda\setminus\mu} = \sum_{\text{SSYT of shape } \lambda\setminus\mu} x^{\mathrm{wt}(T)}$.

Still SSYT are weakly increasin gin rows and strictly in columns.

Then we have two formulas for the skew Schur functions:

Theorem 17.20 (Jacobi–Trudi)

Let $\lambda = (\lambda_1, \ldots, \lambda_n)$ and $\mu = (\mu_1, \ldots, \mu_n)$ (n can be any number greater than or equal to the number of parts in λ and μ; we can append zeros.)

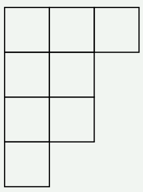
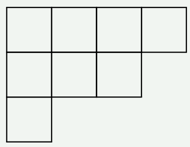
(1) $s_{\lambda/\mu} = \det(h_{\lambda_i - i - \mu_i + j})_{i,j\in[n]}$ (so determinant of a matrix whose entries are complete homogeneous stuff).

(2) It's also $\det(e_{\lambda'_i - i - \mu'_j + j})_{i,j\in[m]}$, where m is anything greater than the number of rows in λ and μ (so in other words $m \geq \max(\lambda_1, \mu_1)$).

Actually we have infintiely amny formual,s since for every n there is a formula.

As usual, $e_0 = h_0 = 1$, and $e_k = h_k = 0$ for $k < 0$.

Example 17.21

Consider the usual shape $\lambda =$  $= (3,2,2,1)$, so $\lambda' =$  $= (4,3,1)$. Then

$$s_{(3,2,2,1)} = \begin{vmatrix} h_3 & h_4 & h_5 & h_6 \\ h_1 & h_2 & h_3 & h_4 \\ h_0 & h_1 & h_2 & h_3 \\ 0 & 0 & 1 & h_1 \end{vmatrix}$$

where indices increase consecutively and our diagonal is our entries.

Similarly we can write it as

$$\begin{vmatrix} e_4 & e_5 & e_6 \\ e_2 & e_3 & e_4 \\ e_{-1} & e_0 & e_2 \end{vmatrix},$$

where we write down our entries on our diagonal and then the stuff in a row are consecutive.

There is a nice combinatorial way to prove both of these formulas. Once we know that they are true, we discussed there is a certain involution ω — switching e's and h's (sending elementary to complete homogeneous and vice versa).

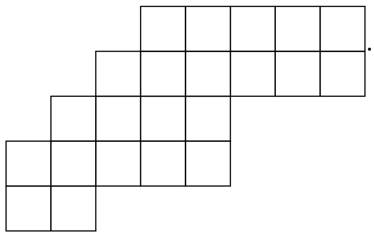
Corollary 17.22

ω sends $s_{\lambda/\mu} \mapsto s_{\lambda'/\mu'}$.

§18 October 19, 2022

§18.1 Jacobi–Trudi Formulas

We saw that if $\lambda = (\lambda_1, \dots, \lambda_n) \supset \mu = (\mu_1, \dots, \mu_n)$, then we could look at the *skew* Young diagram λ/μ. Then take some $m \geq \max(\lambda_1, \mu_1)$, so that our skew shape fits inside a $m \times n$ rectangle.



On one hand we have

$$s_{\lambda/\mu} = \det\left((h_{\lambda_i - i - \mu_j + j})_{i,j \in [n]}\right).$$

Similarly we have

$$s_{\lambda/\mu} = \det\left((e_{\lambda'_i - i - \mu'_j + j})_{i,j \in [m]}\right).$$

Corollary 18.1

The involution $\omega \colon \Lambda \to \Lambda$ sending e to h satisfies $\omega(s_{\lambda/\mu}) = s_{\lambda'/\mu'}$.

Basically, we need to show that conjugation *commutes* with multiplication.

Recall that we know

$$s_\lambda s_\mu = \sum c^\nu_{\lambda\mu} s_\nu,$$

where the $c^\nu_{\lambda\mu}$ are called the **Littlewood-Richardson coefficients**. This property implies that $c^\nu_{\lambda\mu} = c^{\nu'}_{\lambda'\mu'}$. There are several version so fhtis — there are explicit combinatorial rules for these numbers. It may not e totally obvious that these numbers have this symmetry.

There are several ways to formulate the rule; it can be done in such a way that makes it clear, but in th eoriginal way this looks a bit mysterious.

Now we'll prove the Jacobi–Trudi formulas. There's a really nice method of proving this called the Gessel–Viennot method, or the Linström lemma.

§18.2 Linstrom Lemma

Suppose that G is a weighted planar acyclic directed graph. Planar means that the graph can be drawn on the plane; also assume that it not just *can* be drawn on the plane, but it is drawn in the plane.

Acyclic means there are no direct cycles.

Suppose the graph is drawn on teh plane with **sources** $S_1, \dots, S_n$ (special vertices that are marked) on the left sides of the boundary, and *sinks* $T_1, \dots, T_n$ on the right side of the boundary.

$$s_1 \bullet \quad s_2 \bullet \quad s_3 \bullet$$

In between we may have a bunch of vertices and edges, with no crossing edges and no cycles.

We also have *edge weights* — numbers $x_e > 0$ for each edge.

For this graph, we can make a matrix $M = M_G$. This is a $n \times n$ matrix with entries M_{ij}, where

$$M_{ij} = \sum_{P : S_i \to T_j} \mathrm{wt}(P)$$

where the sum is over all directed paths P from S_i to T_j, and $\mathrm{wt}(P)$ is the product of all edge weights on the path.

Lemma 18.2 (Linström Lemma)

$$\det(M) = \sum_{(P_1, \dots, P_n)} \prod_{i=1}^{n} \mathrm{wt}(P_i)$$

where P_i is a path from $S_i \to T_i$ (so we have a n-tuple of paths) and this is actually a n-tuple of *non-crossing paths.*

Here the word non-crossing means that these paths are not allowed to have any common vertex, for any $i \neq j$ — even if we have two paths that touch each other, even if they may not cross, we still consider these as crossing. (So here non-crossing also means non-touching.)

Example 18.3

Consider the graph where we have two sources S_1 and S_2, and T_1 to T_2, and then in the middle we have one edge down and one edge up.

Our little edges on the ends are 1, and in the middle we have x to the right (on S_2 to T_2), y down and z directed up, t directed rightwards (on S_1 to T_2).

Then we get

$$M_G = \begin{bmatrix} t & tz \\ yt & \\ x + ytz & \end{bmatrix}$$

You can easily see that if you take this matrix and take its determinant, then you get some cancellation:

$$\det M_G = xt.$$

We can easily see if we want to connect both sources to both sinks by two paths that don't cross or touch, there's only one way to connect them, one of weight x and one of weight t.

These do not need to be all sources of the graph; sources are just some marked vertices on the left, and sinks some marked vertices on the right.

The condition fo planarity can actually be dropped, and we would get a similar formula. But then in this expressionm when we sum over all n-tuples of paths connecting all sources to all sinks, they may not be connected in the same order. If the graph is planar then S_1 has to be connected to T_1, S_2 to T_2, and so on (planarity forces that the order of the sources is the same as the order of sinks); in general there can be some permutations, and we would need to include the sign of the permutation int he sum. But ing eneral you can write it as an alternating sum over non-crossing path s— that works in a general setting.

But the main thing in this setting is that for planar graphs, this expression here has no minus signs — it is always positive. (In general determinants can have $\pm$, but somehow if the graph is planar, all negative signs cancel and the sum will have positive sign.)

Philosophically, the significance of this lemma is that the notion of pla arity is closely related to the notion of positivity — the fact that the graph is planar means the determinant is positive. In fact, there is something even stronger — there is the notion of a **totally positive matrix** (or totally nonnegative matrix).

Definition 18.4. A $n \times n$ matrix M with real entries is called **totally positive** (or respectively, totally nonnegative) if any minor (determinant of a square submatrix) of M is positive (resp. nonnegative).

Inf act, from the Linstrom lemma, if we have a planar graph adn construct a matrix, then not only is its determinant nonnegative, but *any minor* will be nonnegative.

Corollary 18.5

M_G is totally nonnegative.

To see this, suppose that we want to calculate not just the determinant of this matrix, but some minor of the matrix — so we want to take a subset of rows and a subset of columns, and look at all the matrix entries in these things.

That means you take an arbitrary subset of the sources and an arbitrary subset of sinks (of the same size). But then you can apply the lemma not t oall sources and snks, but these particular subsets; then we get that the minor is a sum over all ways to connect this subset of sources to the subset of sinks. So the lemma implies not only that the entire determinant is nonnegative, but that all minors are.

So planarity is actually closely related to *total positivity* — philosophically tehse things are very close to each other. Hopefully we are going to prove the lemma and Jacobi–Trudi today. But first we will see something not as widely known as the Linstrom lemma: something that Prof. Postnikov thinks of as the *inverse linstrom lemma*.

Lemma 18.6 ("Inverse Linström Lemma")

For any totally nonnegative $n \times n$ matrix M, there eixsts a weighted planar graph (with sources and sinks as here) such that $M = M_G$.

So really, the class of totally nonnegative matrices *is* the class of all matrices tha tcome from planar graphs. This is harder to prove than the Linström lemma.

Student Question. Can mlutpple graph shave the same matrix?

Yes — for example consider $n = 1$, so we just have a nonnegative number. You can get this if you have an arbitrary planar graph with one source adn one sink.

So it si not a 1 to 1 corerespondence. But if you add some addidtional conditions, then you can make it one.

This is closely related to double wiring diagrams.

§18.3 Fomin–Zelevinsky's double wiring diagrams

First let's talk about usual wiring diagrams — a graphical way to show reduced decompositions for S_n.

We start wiht 1234 on teh left and right, and have some wires connecting them.

This is already a graph. But now we are going to transform it into a directed graph of the kind we have.

Basically, for any crossing we are going to split any pair of crossing wires into two vertices.

First we have a crossing between the two bottom wires 1 and 2. We start off with 4 lines directed left to right. Then for the first crossing, we draw a bridge from 1 to 2. Then at the crossing between 1 adn 3 we draw an edge $2 \to 3$. Then we draw an edge $3 \to 4$ for the crossing between 1 (now near the top) an d 4, and then we add another between 3 and 2 (which are now at the bottom, so the endge is $1 \to 2$).

Now for all our horizontal edges, the weight is 1. But our vertical bridges have weighr x_1, x_2, x_3, x_4 from left to eright. Then our corresponding matrix actually factors into four simple, elemnetary factors, called elementary jacobi matrices

$$J_1(x_1)J_2(x_2)J_3(x_3)J_4(x_4)$$

in thsi example. In general, these things are almost the identity mtarix: in one 2×2 block we write

$$\begin{bmatrix} 1 & x \\ 0 & 1 \end{bmatrix}$$

in position i to $i+1$, and everythign else is 1's on the diagonal and 0's off. We can see that every bridge gives you a factor like this.

Theorem 18.7 (Fomin–Zelevinsky)

Any totally nonnegative upper triangular matrix M with 1's on the diagonal comes from this particular class of graphs, corresponding to reduced decompositions of permutations — M can be written as a product of these elementary Jacobi matrices $J_{i_1}(x_1) \cdots J_{i_\ell}(x_{i_\ell})$.

Each of these factors is an upper triangular matrix with 1's on teh diagonal, so the result will also be an upper triangular. So fi we multiply these things, we can't get an arbitrary matrix, butw e can get a totally nonegative upper triangular one.

It's enough to do this for a reduced decomposition, not an arbitrary one. If you want to get a totally *ppositive* matrix, you need to assume that this i sa reduced decomposition for the longest matrix. (Smaller permutations will give you more degenerate cases, where there are zero minors).

Remark 18.8. What is a totally positive upper triangular matrix?

Obviously that cannot happen because it has a lot of 0's. What we actually mean here is that any minor that has a chance to be nonzero is greater than zero — some minors will be identically zero, but totally positive means everythign that oesn't have to be zero is positive.

If you fix the reduced decomposition, then this collection of parametres is unique — there's a unique choicd of eth ex's to produce such a totally positive upper triangular matrix.

Question 18.9. How many minors which are not identically 0 can you find in such a matrix?

Let's start with the case $n = 1$; thenour matrix is $\begin{bmatrix} 1 \end{bmatrix}$. We will say that it has one 1×1 minor, and one empty minor.

Now let's look at the 2×2 case

$$\begin{bmatrix} 1 & * \\ 0 & 1 \end{bmatrix}.$$

This has three 1×1 minors that aren't necessarily 0, so then we get $1 + 3 + 1$ (for 0×0 adn 2×2), so we get 5.

Now for 3×3 we get

$$\begin{bmatrix} 1 & * & * \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This gives $1 + 6 + (9 - 3) + 1 = 14$.

We can guess that these are the Catalan numbers; this is true — for a $n \times n$ matrix we get C_{n+1} minors that are not zero in a generic upper triangular matrix. But this also gives a decmoposition of the Catalan numbers into factors (1, 6, 6, 1). These also have an interesting combinatorial interpretation.

Returnign to Fomin and Zelevinsky's work, this gets upper triangular matrices. Now to get an arbitrary one, we need to consider a slightly more general kind of graph. Every matrix in GL_n can be written as an upper triangular and lower trinagular (LU); this also works for totally nonnegative/positive (any totally nonnegative can be decomposed uniquely as a product of an uppe rtriangular and lower triangular totally positive matrix, and a diagonal thing — LDU decomposition).

For that, we can use these Jacobi matrices and their transposes. This means to get an arbitrary such thing, we need to consider graphs of this form except there are two types of bridges — up-down and down-up bridges. That lets us get an arbitrary totally nonnegative $n \times n$ matrix. Here you are allowed to mix the up and down bridges in any way — so ao double wiring diagram hs tw otypes of crossings, mixed in any way. (One way to make the choice unique is to assume it's a reuced decomposition and all up-nridges go before all downs.)

For the diagonal matrix, we can basically put some edges on the right-hand isde of our horizontla edges — the last horizontal edge in each row (or the first).

§18.4 Back to Lindstrom Lemma

Now let's prove the Lindstrom Lemma. We return to the original setting with an arbitrary graph, and corresonding matrix.

The first step is to write down the definition of the determinant as

$$\det(M_G) = \sum_{w \in S_n} (-1)^{\ell(w)} \prod_{i=1}^{n} M_{i,w(i)}.$$

Again, we can even expand this as

$$\sum_{w \in S_n} (-1)^{\ell(w)} \prod$$

since each M_i is a sum over paths, so we can write their product as a sum over n-tuples of paths

$$\sum_{(P_1,\dots,P_n)} \prod \mathrm{wt}(P_i)$$

where P_i is a path from S_i to $T_{w(i)}$.

Now inside of this huge path, we can do a lot o fcancellations. First, we claim that we can cancel all terms that involve some paths with a crossing; we then want to see that somehow this expressinon is equal to the sum over all non-crossing n-tuples of paths $P_1, \dots, P_n$.

Suppose that we have a pair of paths that cross — for example P_i and P_j; let's draw them with P_j in red, so that they cross. Let's call teh point where they cross c.

So in our n-tuple, if there is one crossing, thenw e can do the following operation (called 'flip the tails') —- we look at the two paths, and we transform them into $\widetilde{P}_i$ and $\widetilde{P}_j$ obtained by flipping the tails, where $\widetilde{P}_i$

starts as P_i until the crossing point, and then continues as the tail of P_j; similarly $\widetilde{P}_j$ starts at P_j up to c, and then continues as P_i.

We don't change any other pairs of paths.

This changes the permutation by a transposition, so its sign changes. But the product of weights is the exact same. SO that means we can kind of cancel the contribution of the first thing with the contribution of the other thing.

That cancellation allows us to get rid of all n-tuples with at least one crossing.

We have left the most important part of this construction under the rug. The tricky part is that if we have only one crossing, it's clear how to cancel. But what if we have a n-tuple with a lot of different crossings?

What we are doing is constructing a sign-reversing involution. There can be many different places where we can apply this operation, and we want to somehow determine the crossing we use in a consistent way, so that if we repeat the thing twice then we go back to the original thing.

So we define c to be the first crossing, but the tricky part is how to consistently define that such that this is actually an involution. This should be lexicographically minimal in some sense, but you have to do it carefully.

§19 October 21, 2022

§19.1 Lindstrom Lemma

If we have a planar acyclic graph G with edge weights, and some marked vertices $S_1, \ldots, S_n$ on the left and sinks $T_1, \ldots, T_n$ on the right, which we are now going to call *targets* (because sources and sinks both start with the letter s), then we can construct a matrix whose (i, j)th entry is the sum of the weights of all paths $p\colon S_i \to T_j$.

Lemma 19.1

The determinant of this matrix is the sum over all n-tuples of *non-crossing paths* connecting all sources with all sinks, of the product of weights.

Proof. First, by expanding the definition,

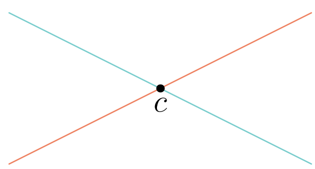
$$\det(M) = \sum_{(P_1,\ldots,P_n)} (-1)^{\ell(w)} \prod \mathrm{wt}(P_i)$$

where $(P_1, \ldots, P_n)$ is an *arbitrary* n-tuple of paths connecting the sources with targets (which may have some intersections) — the sources may go to a permutation of the targets, so we have $P_i\colon S_i \to T_{w(i)}$.

We want to see that many things in this alternating sum can be cancelled, and we'll end up with a sum of non-crossing n-tuples of paths (with no signs).

This involution is based on a very simple operation: if we have a n-tuple of paths with at least one crossing (where two paths have a common point), then we can first find the *first* crossing point (there may be several).

Let's call the crossing paths P_i and P_j, and the crossing c.



Then we want to replace the two paths with $\tilde{P}_i$ and $\tilde{P}_j$, where $\tilde{P}_i$ starts as P_i until the point c, and then continues as P_j after.

This gives a map $\tau : (P_1, \dots, P_n) \to (P_1, \dots, \tilde{P}_i, \dots, \tilde{P}_j, \dots, P_n)$. Then we want τ to satisfy the following properties:

- τ is sign-reversing.
- τ is weight-preserving.
- τ is an involution.

If we can have these properties, then τ will cancel all terms with at least one crossing, and we'll be left with only the non-crossing paths.

The first two properties are obvious — τ multiplies the permutation by a transposition, and it doesn't change the multiset of all edges in the path. So the only nontrivial thing is to make sure this operation is an involution.

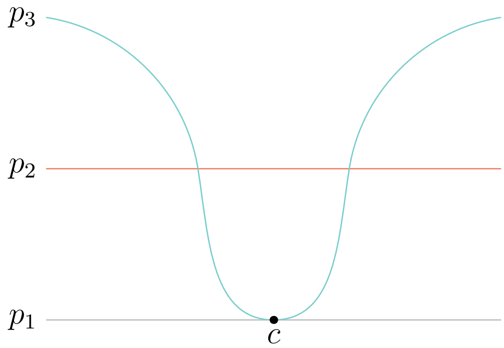
The key point is how to define the 'first crossing point.' First, we need to be careful about what we mean by a crossing. It can't just be a vertex, because there may be other paths through this vertex — and we need to be able to tell which paths we're looking at.

Definition 19.2. A **crossing** is a triple (i, j, c) such that both P_i and P_j contain vertex c.

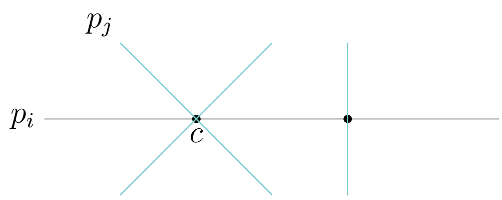
If we have a n-tuple of paths with some crossing, there might be a lot of crossings; how do we find the first one?

One natural attempt to do this is to find the *lexicographically minimal* crossing — we want to find a crossing (i, j, c) that's lexicographically minimal. So we find the first pair of paths (i, j) that have at least one crossing (in terms of lexicographical ordering), and take c to be their first crossing.

But actually, this doesn't work!



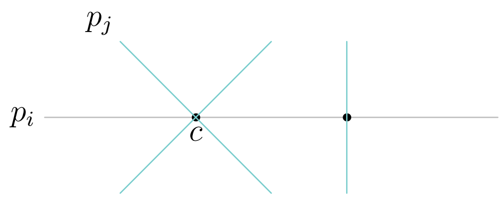
Right now p_1 and p_2 don't intersect, so our lexicographically minimal crossing is c. But if we swap, then we get that $\tilde{p}_1$ and p_2 *do* intersect! So the point we're swapping at is the new point c' to the right, which isn't what we want.



Now let's look at a method which *does* work — first, we find the minimal i such that P_i has a common point with another P_j.

In our failed method, we tried to minimize i, then j, then c. But instead, we can first try to minimize i, then c, then j — so we're taking a lexicographically minimal point, but in a slightly different order.

So we find i, and then we find the first vertex c in p_i that also belongs to another path p_j. Now we want to minimize j — so now we find the minimal $j \neq i$ such that $c \in p_j$.



Example 19.3

In our three-path example from earlier, we start off by doing the same thing. But then we find the minimal path that has a crossing — that's p_1 — and then we find the first piont on p_1 that has a crossing — that's c — and then we find the first path through it. So we do get back our original paths.

Remark 19.4. There are other methods as well. For example, we can order all vertices in some way, find the minimal vertex c involved in a crossing, and then find the lexicographically minimal pair $i \neq j$ such that p_i and p_j contain c.

This involution now cancels out all n-tuples of paths with at least one crossing, so we're left with only the ones with no crossings. □

§19.2 Jacobi–Trudi Formulas

Now let's return to the Jacobi–Trudi formulas:

Theorem 19.5

We have the following two formulas:

$$s_{\lambda/\mu}(x_1, \dots, x_N) = \det\left(h_{\lambda_i - i - \mu_j + j}(x_1, \dots, x_N)_{i,j \in [n]}\right),$$

and similarly

$$s_{\lambda/\mu}(x_1, \dots, x_N) = \det\left((e_{\lambda'_i - i - \mu'_j + j}(x_1, \dots, x_N)_{i,j \in [m]}\right),$$

where n is at least the number of parts of λ and μ, and m at least the maximal part.

Last class, we formulated the problem in terms of symmetric *functions*, in infinitely many variables; but now we'll formulate it in terms of symmetric *polynomials* in any arbitrary number of variables N (which is unrelated to the number of parts n of our permutations).

Remark 19.6. The two versions are equivalent — we can specialize the infinite version to the finite version, or take the limit to go from the finite version to the infinite one. But the proof is slightly easier for the finite version.

The main idea of the proof is to apply the Lindstrom lemma on a certain graph. The graph we'll use is a *square grid* graph, or rather a *strip* of the square grid of height $N-1$. All edges are directed from left to right, and from bottom to top.

We place weights on these edges, where the weights of all vertical edges are 1, and the weights of all horizontal edges on level i are x_i. So on the bottom row, all horizontal edges have weight x_1; on the second, all edges have weight x_2; and so on, up to the top row where all horizontal edges have weight x_N. (Meanwhile all vertical edges have weight 1.)

Now we need to define our sources. In this case, we'll place the sources on the bottom and the targets on the top. We place sources s_1, s_2, $\ldots$, s_n on the bottom, and t_1, t_2, $\ldots$, t_n on the top. The sources are defined in terms of the parts of μ — we have $s_i = (\mu_i + n - i, 0)$, and $t_j = (\lambda_j + n - j, N-1)$. So the x-coordinates of sources and targets are the parts of λ and μ, but shifted a bit — instead of weakly decreasing partitions, we make a strict partition by adding the extra numbers $n-i$.

First let's calculate M_{ij}. This means we want to sum over all paths from s_i to t_j the product of edge weights. In any lattice path from s_i to t_j, we have some number of steps to the right, with weight x_1; then we take an up-step; then we take some more number of steps to the right, with weight x_2, and then take one step up; and so on. If we take two steps right, then one up, then oen right, then two up, then one right, then two up, then one right, then we get $x_1^2 x_2 x_4 x_6$.

In general, it's easy to see that

$$M_{ij} = h_k(x_1, \ldots, x_N),$$

where k is the horizontal distance between s_i and t_j — so $k = \lambda_j - j - \mu_i + i$. So M is exactly the transpose of the Jacobi–Trudi matrix for $s_{\lambda/\mu}$.

Now we've seen that the right-hand side of the Lindstrom lemma is equal to the right-hand side of the Jacobi–Trudi formula; now let's see that the left-hand sides are also equal.

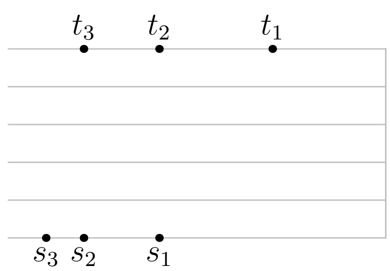
Lemma 19.7

There is a bijection between n-tuples of non-crossing paths $(p_1, \ldots, p_n)$ for this graph, and semistandard Young tableaux of shape λ/μ such that the weight given by the n-tuple of paths is exactly the weight of the corresponding SSYT.

We'll prove this by example, as always.

Example 19.8

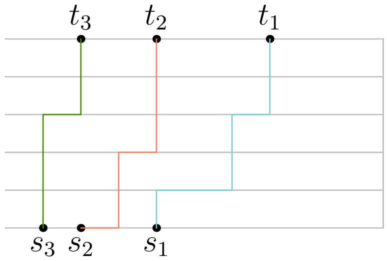
Take $\lambda = (5, 3, 2)$ and $\mu = (2, 1, 1)$, and $N = 6$.



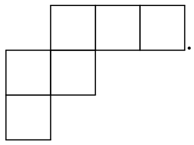
First, s_1 is given by the first part of μ. Then the second part of μ is given by 1 minus the first part, so we skip 2 steps to the left.

We can do the same for λ and the t_i — we start with t_1 three right of s_1, then skip three steps, then two.

Now we need to choose a set of non-crossing paths:



Meanwhile, we have

$\lambda/\mu =$ .

Given this set of paths, we'd now write down

	2	2	4
1	3		
4			

We take the first path and write down its weight entries in the first row, then the second path's entries in the second, and so on.

It's clear entries are nondecreasing in rows. Meanwhile, to show that they're strictly increasing in columns, suppose that we have a directly above b. Then we can find x_a on our first path. Meanwhile, we claim that b comes from the first edge whose right endpoint is directly above the left endpoint of a's edge. Clearly since the two paths are non-crossing, we must have $a < b$.

On the other hand, if we carefully look at this, then these conditions are enough — if we have any semi-standard Young tableau, we can convert it into a collection of paths. Then our collection of paths must satisfy this condition, and it's a little combinatorial exercise to show that if this condition holds (whenever we have a pair of horizontal edges in two adjacent paths, the edge in the right path is strictly below the edge in the left path), then the paths are non-crossing.

Finally, let's prove the second part of the Jacobi–Trudi formula. We essentially do the same thing, but we have to slightly modify the graph.

Now instead of horizontal strips in the square grid, we look at a *diagonal* part of the square grid — we put our sources on one diagonal, and our targets on the opposite diagonal, where the distance between the two diagonals is again $N-1$. All edges are still directed left to right and bottom to top. The weights on vertical edges are still 1. Meanwhile, on the horizontal edges, we have $x_1, x_2, \ldots, x_N$ on each horizontal line — so i is not the *height* of the edge, but the diagonal containing the edge.

Then applying the same argument to this graph, we'll see that weights correspond to the elementary symmetric polynomials and the matrix M_{ij} corresponding to this graph is the right-hand side of the Jacobi–Trudi formula, and the same Lindstrom lemma argument (considering non-crossing paths) proves the equality.

§20 October 24, 2022

Last time, we discussed the two Jacobi–Trudi formulas. Today we'll talk about applications of these two formulas.

One application is that if ω is the automorphism on the ring of symmetric functions sending $e_k \leftrightarrow h_k$, then $\omega(s_{\lambda/\mu}) = s_{\lambda'/\mu'}$.

§20.1 Combinatorial and Classical Definitions of Schur Functions

Now we'll use this to prove the following (which we've mentioned earlier):

Theorem 20.1

We have $s_\lambda^{\text{comb}} = s_\lambda^{\text{class}}$.

The Jacobi–Trudy formula gives us a determinental expression for the left-hand side. Meanwhile, the right-hand side is defined as a quotient of two determinants. So we want to show that one determinant equals the quotient of two determinants.

However, these determinants are quite different — on the left-hand side we have the Jacobi–Trudy matrix, while on the right-hand side we have the Vandermonde determinant and a Vandermonde-like matrix.

Proof. We define three $n \times n$ matrices, indexed by $i, j \in [n]$:

- The matrix $A_\alpha = (x_j^{\alpha_i})$ for $i, j \in [n]$, where α is any nonnegative integer vector $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{Z}_{\geq 0}$.
- The matrix $H_\alpha = (h_{\alpha_i - n + j}(x_1, \dots, x_n))$, where n is the number of rows in our skew shape.
- The matrix $E = ((-1)^{n-i} e_{n-i}^{(j)})$, where $e_k^{(j)}$ is the elementary symmetric function $e_k(x_1, \dots, \hat{x}_j, \dots, x_n)$, where we *skip* the jth variable.

Lemma 20.2

These matrices are related to each other by $A_\alpha = H_\alpha \cdot E$.

Proof. One way to see this is to write down generating functions — consider

$$E^{(j)}(t) = \sum_{k \geq 0} e_k^{(j)} t^k.$$

Earlier we saw how to write the generating function for the usual elementary symmetric function, and now we simply skip the jth term, so we write

$$E^{(j)}(t) = (1 + x_1 t)(1 + x_2 t) \cdots \widehat{(1 + x_j t)} \cdots (1 + x_n t).$$

Similarly, we can take

$$H(t) = \sum_{k \geq 0} h_k(x_1, \dots, x_n) t^k = \frac{1}{1 - x_1 t} \cdot \frac{1}{1 - x_2 t} \cdots \frac{1}{1 - x_n t}.$$

to be the usual generating function for complete homogeneous polynomials, in all n variables. Multiplying these together, we see that

$$H(t) \cdot E^{(j)}(-t) = \frac{1}{1 - x_j t}.$$

Now we can extract the coefficient of a given power of t on both sides.

Notation 20.3. We use $[t^{\alpha_i}](\cdots)$ to denote the coefficient of t^{α_i} in some given power series.

On the left-hand side, we have

$$[t^{\alpha_i}](\text{LHS}) = \sum_{a+b=\alpha_i} h_a \cdot (-1)^b e_b^{(j)}.$$

(Here a and b are just integer indices, unrelated to the entries of A.) On the other hand, the coefficient on the right-hand side is

$$[t^{\alpha_i}](\text{RHS}) = x_j^{\alpha_i}.$$

Now $x_j^{\alpha_i}$ is the (i,j)th entry of A_α, and we claim that the first expression is the (i,j)th entry of $H_\alpha \cdot E$. To see this, rename the variables to $b = n-k$, so that $a = \alpha_i - b = \alpha_i - n + k$. Then we can see immediately that our expression h_a will be the (i,k)th entry of the matrix H_α, and $(-1)^b e_b^{(j)}$ will be the (k,j)th entry of the matrix E; then summing over all k gives us exactly the (i,j)th entry of $H_\alpha \cdot E$. $\square$

Remark 20.4. There should also be a nice combinatorial way to see this identity.

Now to prove our theorem, by definition we know

$$a_\alpha = \det A_\alpha = \det(H_\alpha)\det(E)$$

(here a_α is the determinant *defined* as $\det A_\alpha$ from earlier). Then we can take the staircase partition $\delta = (n-1, n-2, \dots, 0)$, so that a_δ is the usual Vandermonde determinant. In this case, H_α is an upper triangular matrix with 1's on the diagonal, so we get that

$$a_\delta = 1 \cdot \det(E).$$

Now taking λ to be a partition, we have

$$a_{\lambda+\delta} = \det(A_{\lambda+\delta}) = \det(H_{\lambda+\delta}) \cdot \det(E).$$

But we know that $\det(E) = a_\delta$, so then we can conclude that

$$\frac{a_{\lambda+\delta}}{a_\delta} = \det(H_{\lambda+\delta}).$$

But the left-hand side is our classical definition of Schur functions, and the right-hand side is the expression for combinatorial Scuhr functions given by the Jacobi–Trudi formulas. So then this gives that $s_\lambda^{\text{class}} = s_\lambda^{\text{comb}}$. $\square$

§20.2 Determinental Formula for the Number of SYT

Notation 20.5. Let $f^{\lambda/\mu}$ be the number of SYT of skew shape λ/μ.

If μ is empty, then we get straight shapes, and f^λ is given by the hooklength formula. But if we don't have a straight shape, we may not get a product formula. However, we *can* get the following theorem (independently proved by Frobenius and Young around 1900):

Theorem 20.6 (Frobenius and Young)
We have

$$f^{\lambda/\mu} = N! \cdot \det\left(\frac{1}{(1-\lambda_i-\mu_j+j)!}\right)_{i,j\in[n]}$$

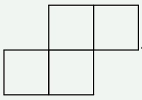
where N is the number of boxes in λ/μ, and n is the number of rows.

Remark 20.7. This formula was proved almost half a century before the hook-length formula, and the original proof of the hook-length formula came from this expression (performing some matrix manipulations to show that the determinant is given by a simple product).

Note that here we have a matrix, whose entries are of the form $1/x!$. Our conventions are that $0! = 1$ and for $k > 0$, $1/(-k)! = 0$. This kind of makes sense — it says that $(-k)!$ should be infinity, which makes sense because $(-k)! = \Gamma(-k+1)$, and $\Gamma(z) = \int_0^\infty t^{z-1}e^t\,dt$ (a continuous generalization of the factorial, which satisfies $n! = \Gamma(n+1)$) has poles at 0, -1, -2, and so on; so it is natural to assume that negative factorials are infinity.

Example 20.8

Suppose we take $\lambda = (3,2)$ and $\mu = (1,0)$, so

$\lambda/\mu =$ .

Here $n = 2$ and $N = 4$. In this case, this formula tells us

$$f^{\lambda/\mu} = 4! \cdot \begin{vmatrix} \frac{1}{2!} & \frac{1}{4!} \\ \frac{1}{0!} & \frac{1}{2!} \end{vmatrix} = 5.$$

We can check that there are exactly 5 SYT of this shape.

On the other hand, we can also apply a similar formula to the *conjugate* partition:

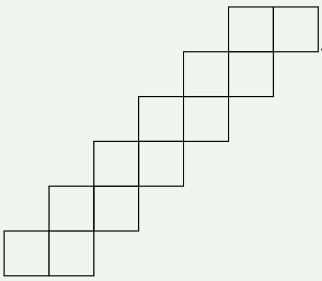
Example 20.9

We have $\lambda' = (2,2,1)$ and $\mu' = (1,0,0)$, so now $m = 3$ (where m is the number of rows in the conjugate, or equivalently the number of columns in the original). This gives us an expression for the same quantity in terms of a 3×3 matrix (since $f^{\lambda/\mu} = f^{\lambda'/\mu'}$ — the number of SYT won't change when we conjugate): we have

$$f^{\lambda'/\mu'} = 4! \begin{vmatrix} \frac{1}{1!} & \frac{1}{3!} & \frac{1}{4!} \\ 1 & \frac{1}{2!} & \frac{1}{3!} \\ 0 & 1 & \frac{1}{1!} \end{vmatrix}.$$

As a digression, the number 5 that we got here belongs to an interesting sequence of numbers. It also belongs to a lot of sequences — the Fibonacci numbers, natural numbers, and Catalan numbers, but Prof. Postnikov has a different number in mind.

Question 20.10. How many SYT does the 'zigzag ribbon' shape have?

.

Thinking of our SYT as a permutation of $[N]$, we obtain the description

$$A_N := \#\{w \in S_N \mid w_1 < w_2 > w_3 < w_4 > \cdots\}.$$

These numbers are called the *zigzag numbers*, or *up-down* numbers. They're also called Andre numbers, Euler numbers, tangent and secant numbers, and various other names.

These numbers have a lot of interesting properties, but one of the most famous is the following:

Theorem 20.11 (André Theorem)

The exponential generating function for these numbers is given by

$$\sum_{N \geq 0} A_N \cdot \frac{x^N}{N!} = \tan(x) + \sec(x).$$

Since $\tan(x)$ is odd and $\sec(x)$ is even, this means the even terms are given by the expansion of sec, and the odd terms by tan. The first few terms (starting with 0) are $1, 1, 1, 2, 5, 16, 61, 272, \ldots$.

Meanwhile, our formula gives us a way to write A_N as $N!$ times the determinant of some $(N/2) \times (N/2)$ matrix. For even N, we can write this down explicitly:

Corollary 20.12

We have

$$A_N = N! \cdot \begin{bmatrix} \frac{1}{2!} & \frac{1}{4!} & \frac{1}{6!} & \cdots & & \\ 1 & \frac{1}{2!} & \frac{1}{4!} & \frac{1}{6!} & \cdots & \\ 0 & 1 & \frac{1}{2!} & \frac{1}{4!} & \frac{1}{6!} & \cdots \\ 0 & 0 & 1 & \frac{1}{2!} & \frac{1}{4!} & \cdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \\ \vdots & & & & & \end{bmatrix}$$

Proof of Theorem. By Jacobi–Trudi, we know

$$s_{\lambda/\mu} = \det((h_{\lambda_i - i - \mu_j + j})),$$

while this formula tells us

$$f^{\lambda/\mu} = N! \cdot \det\left(\frac{1}{(\lambda_i - i + \mu_j - j)!}\right).$$

We'd like to see how these are related.

First, we want to see how we can use $s_{\lambda/\mu}$ to calculate the number of *standard* Young tableau (they're defined according to *semistandard* Young tableau).

Lemma 20.13

We have $f^{\lambda/\mu} = [x_1 \cdots x_N](s_{\lambda/mu})$.

This makes sense — the coefficient on the right-hand side gives the number of semistandard Young tableau with weight $(1, 1, \ldots, 1)$, which are exactly standard Young tableau.

So we'd like an operation that extracts this coefficient. That operation is given by *exponential specialization* of symmetric functions: the map $\mathrm{ex}\colon \Lambda_{\mathbb{C}} \to \mathbb{C}[t]$ (this works over any field of characteristic 0, not just $\mathbb{C}$) defined as follows:

First, the fundamental theorem of symmetric functions says that the ring of symmetric functions is a ring of polynomials in e_1, e_2, e_3, ... We can use the p-version of this theorem — a similar theorem in terms of power symmetric functions.

Theorem 20.14

Let p_k be the *power symmetric function* $p_k = x_1^k + x_2^k + x_3^k + \cdots$. Then $\Lambda_{\mathbb{C}} = \mathbb{C}[p_1, p_2, p_3, \ldots]$ — every symmetric function can be expressed uniquely as a polynomial in the power symmetric functions.

There is one difference to the version of this theorem for e or h — those theorems work over the integers as well, but here you do need a *field* (you may have to divide).

Now we define the exponential specialization as follows:

Definition 20.15. For a symmetric function $g(p_1, p_2, \ldots)$, we define the map $\mathrm{ex}\colon g(p_1, p_2, \ldots) \mapsto g(t, 0, 0, \ldots)$.

Example 20.16

We have $\mathrm{ex}(e_1) = \mathrm{ex}(p_1) = t$.

Example 20.17

To find $\mathrm{ex}(e_2)$, we have $p_1^2 = x_1^2 + x_2^2 + \cdots + 2x_1x_2 + \cdots = p_2 + 2e_2$, so we can write $e_2 = (p_1^2 - p_2)/2$. This means

$$\mathrm{ex}(e_2) = \mathrm{ex}\left(\frac{p_1^2 - p_2}{2}\right) = \frac{t^2}{2}$$

(replacing p_1 by t and p_2 by 0).

Lemma 20.18

For all symmetric functions f, we have

$$\mathrm{ex}(f) = \sum_{N \geq 0} [x_1 \cdots x_N](f) \cdot \frac{t^N}{N!}.$$

This can be proven by checking it for the product of power symmetric functions. Then applying this to both sides of Jacobi–Trudy, the Schur functions become the numbers of SYT and the determinant on the right-hand side becomes the reciprocal of factorials. $\square$

§21 October 31, 2022

Today (and in the next few lectures) we will discuss representation theory. We may or may not know representation theory, so today we will see a mild introduction; if we have heard nothing about representations, that is perfectly fine.

§21.1 Representations of Finite Groups

Suppose that G is a finite group. In this course, we will mostly be interested in the symmetric group S_n, but for now we'll discuss this in terms of an arbitrary group.

When we talk about representations, we need to fix a field. To keep things simple, we will assume our field is $\mathbb{C}$ (because this is a simpler case).

Definition 21.1. A **representation** of G over $\mathbb{C}$ is a homomorphism $\rho: G \to \mathrm{GL}(V)$, for some vector space V over $\mathbb{C}$.

We will only discuss *finite-dimensional* representations, where V is finite; so $V \cong \mathbb{C}^d$.

In a more elementary interpretation, if we pick a basis for V, then $\mathrm{GL}(V)$ becomes a group of *matrices*. So for every element of our group, we're assigning it a $d \times d$ matrix, such that taking the product of two elements of the group corresponds to taking the product of the matrices.

Of course, there are many representations. In fact, there are many *linear spaces*. But of course we only consider linear spaces up to isomorphism — up to isomorphism there's only one one-dimensional linear space.

Remark 21.2. The cardinality of the set of all one-dimensional linear spaces over $\mathbb{C}$ doesn't make sense — there is no such thing as the set of linear spaces. (This is not a set; so you cannot say there's infinitely many or continuum many, it's just not a set.)

Similarly, when we talk about representations we want to only consider them up to isomorphism.

Notation 21.3. When we talk about representations, we often just denote them by V (the same letter as the linear space) — we think of the representation as our linear vector space with an action of the group.

Definition 21.4. Two representations $\rho_1: G \to \mathrm{GL}(V)$ and $\rho_2: G \to \mathrm{GL}(V_2)$ are **isomorphic** if there exists a linear isomorphism $A: V_1 \to V_2$ such that $\rho_1(g) = A\rho_2(g)A^{-1}$ for all $g \in G$.

In other words, we can identify our two linear spaces so that the action of the group is the same. We will only consider representations up to isomorphism.

Definition 21.5. A representation $\rho: G \to \mathrm{GL}(V)$ is **irreducible** if there is no proper subspace $W \subset V$ with $W \neq \{0\}$ which is invariant under the action of G — in other words, such that $\rho(g): W \to W$ for all $g \in G$.

So we can't find a smaller subspace of our linear space such that all elements of the group preserve it.

The primary problem of representation theory is to classify irreducible representations (up to isomorphism). There are two trivial operations on representations, called the *direct sum* and *tensor product*.

Definition 21.6. Suppose $\rho_1: G \to \mathrm{GL}(U)$ and $\rho_2: G \to \mathrm{GL}(V)$ are representations. Then their **direct sum** $\rho_1 \oplus \rho_2$ is the map $G \to \mathrm{GL}(U \oplus V)$

If we think of $\mathrm{GL}(U)$ and $\mathrm{GL}(V)$ as groups of matrices (fixing a basis), then when we take direct sums, we're taking the *block matrix*

$$\left[\begin{array}{c|c} \rho_1(g) & 0 \\ \hline 0 & \rho_2(g) \end{array}\right].$$

Definition 21.7. The **tensor product** is the representation $\rho_1 \otimes \rho_2: G \to \mathrm{GL}(U \otimes V)$.

(We will not use this today.)

We'll now list (without proof) some facts about representations:

Theorem 21.8

For a finite group, there are finitely many isomorphism classes of irreducible representations. More specifically, the number of irreducible representations equals the number of conjugacy classes of G.

Theorem 21.9

Any representation of G decomposes into a direct sum of irreducible representations (Maschke's Theorem).

This means our main problem is to classify the *irreducible* representations.

Notation 21.10. We use $G^\wedge = \{V_1, \ldots, V_p\}$ to denote the set of irreducible representations of G, up to isomorphism (where p is the number of conjugacy classes in G). We also use $d_i = \dim V_i$ (over $\mathbb{C}$).

There is a fundamental isomorphism, which will be very important for us:

Definition 21.11. Let $\mathbb{C}[G]$ be the group algebra of the group G — the space of linear combinations of group elements, where we multiply elements in the expected way.

Theorem 21.12

As an algebra, $\mathbb{C}[G] \cong \bigoplus_{i=1}^p \operatorname{End}(V_i)$.

Definition 21.13. $\operatorname{End}(V_i)$ is the algebra of endomorphisms of V_i — the set of all linear operators from $V_i \to V_i$ (or equivalently $d_i \times d_i$ matrices).

So in a clearer way, the theorem says that the group algebra $\mathbb{C}[G]$ is isomorphic to the algebra of block-diagonal matrices, where the sizes of the blocks are the dimensions of the irreducible representations.

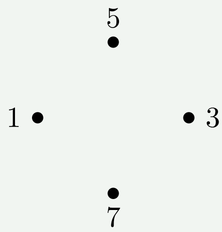
§21.2 Representations of S_n

We'll now assume that $G = S_n$. First we'll try to figure out the number of conjugacy classes.

For two permutations $u, v \in S_n$, we know u is conjugate to v (which we write as $u \sim v$) if there exists a permutation $w \in S_n$ such that $v = wuw^{-1}$. In S_n, this is equivalent to saying that u and v have the same *cyclic type*.

Example 21.14

Consider the permutation $u = (1537)(26)(4)$, and represent it as a diagram:



We can see that u decomposes as a 4-cycle, 2-cycle, and 1-cycle; we call that its *cyclic type*.

Now suppose we had a different permutation v with the same cycle type — for example, we could take $v = (3624)(17)(5)$.

We can then take w to be the permutation

$$w = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 1 & 2 & 5 & 6 & 7 & 4 \end{bmatrix}.$$

Then conjugating by w relabels the elements.

This means conjugacy classes are described by the cyclic type. In particular, the number of conjugacy classes in S_n is $p(n)$, the number of partitions of n — or equivalently, the number of Young diagrams with n boxes.

So it's natural to assume that irreducible representations should be labeled by Young diagrams of size n.

§21.3 The Classical Construction

This construction is called the **Young symmetrizer**. For every Young diagram, we want to construct an irreducible representation. We do this in the following way:

First pick $\lambda \vdash n$, and pick any tableau of shape λ. Here there should be exactly one entry of $1, 2, 3, \ldots, n$, but they don't have to be increasing in rows and columns — for example, we could have

$$T = \begin{array}{|c|c|c|} \hline 4 & 2 & 3 \\ \hline 1 & 5 & \\ \cline{1-2} \end{array}.$$

Now we're going to construct a certain element of the group algebra:

Let

$$a_T = \sum u$$

where the sum is over all permutations $u \in S_n$ that preserve the rows of T, and

$$b_t = \sum (-1)^{\ell(v)} v$$

where the sum is over all $v \in S_n$ that preserve the columns of T; finally let

$$c_T = a_T b_T \in \mathbb{C}[S_n].$$

Then we define

$$V_T = \mathbb{C}[S_n] \cdot c_T$$

to be a certain subspace of the group algebra. On this subspace, S_n acts by left multiplication.

This gives a representation of S_n.

Fact 21.15 — $V_T \cong V_{T'}$ if and only if T and T' have the same shape.

If we took the same Young diagram but put different labels, then we'd get different elements a_T, b_T, and c_T, so we'd get a different subspace of $\mathbb{C}[S_n]$ — but this would be isomorphic.

This means we can now use V_λ to denote V_T for any T of shape λ.

Fact 21.16 — We have $S_n^\wedge = \{V_\lambda \mid \lambda \vdash n\}$.

This is the classical construction.

Student Question. *What is the homomorphism between V_T and $V_{T'}$?*

Answer. Suppose for example that

$$T = \begin{array}{|c|c|c|}\hline 4 & 2 & 3 \\ \hline 1 & 5 \\ \cline{1-2}\end{array} \text{ and } T' = \begin{array}{|c|c|c|}\hline 5 & 1 & 3 \\ \hline 2 & 4 \\ \cline{1-2}\end{array}.$$

Then a_T will be the sum of all permutations that permute $\{4, 2, 3\}$ and $\{1, 5\}$. Meanwhile, b_T will be the set of the four permutations that preserve the columns (we can either leave 4 and 1 or switch them, and either leave 2 and 5 or switch them). Then we take the product of a_T and b_T in the group algebra, and take the left action of the symmetric group.

If we do the same for T', then we get two different subspaces; but one can be obtained from the other by the right-action of the permutation sending $4 \to 5$, $2 \to 1$, $3 \to 3$, and so on.

Example 21.17

Construct all representations of S_3 using this construction.

Solution. First let's start with $\begin{array}{|c|c|c|}\hline & & \\ \hline\end{array}$, and take any tableau, for example $\begin{array}{|c|c|c|}\hline 2 & 3 & 1 \\ \hline\end{array}$. Then any permutation preserves the rows of our tableau, so

$$a_T = 1 + s_1 + s_2 + s_1 s_2 + s_2 s_1 + w_0.$$

Meanwhile, $b_T = 1$. We then have $c_T = a_T b_T$.

We can think of $\mathbb{C}[S_3]$ as the space of linear combinations of our six permutations; this means it's isomorphism to $\mathbb{C}^6$, and c_T would correspond to $(1, 1, 1, 1, 1, 1)$.

Now we want to consider $\mathbb{C}[S_3]c_T$, the subspace spanned by $w(1 + s_1 + s_2 + s_1 s_2 + s_2 s_1 + w_0)$. Of course, if we multiply this permutation by any vector, we get back the same vector. So then $\mathbb{C}[S_3]c_T$ is the one-dimensional subspace consisting of $c(1 + s_1 + s_2 + \cdots + w_0)$.

If we take any permutation and act by this vector on the left, then we get back the same vector. This gives the **trivial representation** — it's one-dimensional, and every element acts by 1.

Now we can do the same thing for the column

$$\lambda = \begin{array}{|c|}\hline \\ \hline \\ \hline \\ \hline\end{array} \rightsquigarrow \begin{array}{|c|}\hline 1 \\ \hline 3 \\ \hline 2 \\ \hline\end{array}.$$

Then only the identity preserves all rows, so $a_T = 1$. Meanwhile b_T is the alternating sum of all permutations, so

$$b_T = 1 - s_1 - s_2 + s_1 s_2 + s_2 s_1 - w_0.$$

Then our subspace is the span of terms $w(1 - s_1 - s_2 + s_1 s_2 + s_2 s_1 - w_0)$. The only things we can produce by multiplying by w are this vector or its negative, so then

$$V_{\begin{array}{|c|}\hline \\ \hline \\ \hline \\ \hline \end{array}} = \{c(1 - s_1 - s_2 + s_1 s_2 + s_2 s_1 - w_0)\}\,.$$

Now if we multiply by any permutation w on the left, if w is even then we get our same vector back, while if w is odd then we get back its negative. This gives the one-dimensional **sign representation**, where w acts as $(-1)^{\ell(w)}$.

The only nontrivial case is when we have

$$\begin{array}{|c|c|}\hline & \\ \hline & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \rightsquigarrow \begin{array}{|c|c|}\hline 2 & 3 \\ \hline 1 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}.$$

Then there's two permutations that preserve rows, so

$$a_T = 1 + s_2.$$

Meanwhile there's only two permutations that preserve columns, so

$$b_T = 1 - s_1.$$

Then we have

$$c_T = (1 + s_2)(1 - s_1) = 1 - s_1 + s_2 - s_1 s_2.$$

Now we want to take this vector and consider V_T, the subspace of $\mathbb{C}[S_3]$ spanned by elements of the form $w(1 - s_1 + s_2 - s_1 s_2)$. *A priori* this could produce six vectors, but we claim that we only get a 2-dimensional subspace: we can represent

$$1 - s_1 + s_2 - s_1 s_2 \rightsquigarrow (1, -1, 1, -1, 0, 0).$$

If we multiply by s_1 on the left, then we get

$$s_1 - 1 + s_1 s_2 - s_2 \rightsquigarrow (-1, 1, -1, 1, 0, 0),$$

which is the negative of the same vector. If we multiply by s_2 on the left, then we get

$$s_2 - s_2 s_1 + 1 - s_2 s_1 s_2 \rightsquigarrow (1, 0, 1, 0, -1, -1).$$

We can continue doing this, and we'll get six different vectors, but they span a 2-dimensional subspace, so our representation is two-dimensional. □

Remark 21.18. It's possible to write everything up explicitly, but it's a bit complicated; if we do all these calculations in terms of the group algebra, they become a bit messy. One thing that's not completely satisfactory about this construction is that it's not natural how you get all the combinatorics we're talking about from scratch. Next lecture we'll try to arrive at the same result but without making any assumptions — that the representations should be labelled with Young diagrams and so on. This comes from trying to construct representations of S_n without making any *a priori* assumptions; the key idea of Vershik and Okounkov is that it's not good to consider just one particular symmetric group, but we instead should consider *all* symmetric groups at once — the chain $S_1 \hookrightarrow S_2 \hookrightarrow S_3 \hookrightarrow \cdots$. If we look at all of these at once, then there's a way to construct all the representations and see how they're related to the combinatorics we've discussed in this class so far.

§22 November 2, 2022

Last time, we mentioned a few basic facts about representations (of finite groups and over complex numbers).

Notation 22.1. For a finite group G, we use $G^\wedge$ to denote the set $\{V_1, \dots, V_N\}$ of irreducible representations of G, and d_i to denote the dimension of V_i.

Theorem 22.2

Every representation V of G is isomorphic to a direct sum

$$V = \underbrace{V_1 \oplus \cdots \oplus V_1}_{n_1} \oplus \underbrace{V_2 \oplus \cdots \oplus V_2}_{n_2} \oplus \cdots = V_1^{n_1} \oplus V_2^{n_2} \oplus \cdots.$$

In general, there might be several different ways to break V into a direct sum (although the dimensions only depend on V). But it turns out the following is true:

Fact 22.3 — If all n_i are 0 or 1 (the **multiplicity-free** case) then this decomposition is unique.

Last time, we also mentioned the *group algebra*:

Theorem 22.4

For any finite group, the group algebra $\mathbb{C}[G]$ is isomorphic to the algebra of block-diagonal matrices where the blocks have sizes $d_1, \dots, d_N$.

In particular, comparing dimensions gives the following:

Corollary 22.5

We have

$$|G| = \sum_{i=1}^{N} d_i^2.$$

In the case of S_n, irreducible representations correspond to Young diagrams of size n, and the dimension is the number of SYT; this gives the identity

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

Last class, we also mentioned the *classical construction* of irreducible representations of S_n — given λ we take any tableau T (not necessarily increasing in rows and columns), and we define $V_T = \mathbb{C}[S_n] a_T b_T$. We can also define another representation on

$$widetildeV_T = \mathbb{C}[S_n] b_T a_T.$$

(Recall that a_T is the sum over all permutations that fix the rows of λ, and b_T the alternating sum over permutations that fix columns.)

Fact 22.6 — $V_T \cong \widetilde{V}_T$.

Question 22.7. Can we see this combinatorially? In other words, find two invertible elements $f, g \in \mathbb{C}[S_n]$ such that $a_T \cdot b_T = f \cdot b_T a_T g$.

From general theory, it follows that f and g should exist; but all proofs that Prof. Postnikov knows of are very indirect.

So the group algebra of S_n is very easy to define, but highly nontrivial.

§22.1 Vershik–Okounkov's New Approach

We'll now discuss Vershik–Okounkov's new approach to representations of S_n.

In the classical construction, we start from combinatorics — we define Young diagrams and Young tableaux and the Young symmetrizer, and then we do some things to show that they correspond to representations. Vershik and Okounkov wanted to construct representations in the opposite way — to get all this combinatorics intrinsically from algebra, in a more direct way. The goal is to construct representations without *a priori* knowing anything about Young tableaux.

In the classical approach, we treat each symmetric group individually. But in a sense, it's more correct to look at all symmetric groups at once — to study the entire chain

$$S_0 \subset S_1 \subset S_2 \subset \cdots.$$

Here S_n is embedded into S_{n+1} in the standard way — if we have a permutation w, we embed it as

$$w = \begin{bmatrix} 1 & 2 & \cdots & n \\ w_1 & w_2 & \cdots & w_n \end{bmatrix} \mapsto \begin{bmatrix} 1 & \cdots & n & n+1 \\ w_1 & \cdots & w_n & n+1 \end{bmatrix}.$$

Exercise 22.8. Find an example of a way to embed S_n into S_{n+1} that is not isomorphic to this one. (Of course we could say 'fix the minimal element' instead, which is isomorphic, but it turns out there are other ways to do this embedding as well.)

The whole theory will depend on this sequence of embedded groups. The crucial fact is the following: we know that $S_n^\wedge = \{V_\lambda \mid \lambda \vdash n\}$, but right now let's not assume λ is a Young diagram, and just think of them as formal labellings.

If we have a group representation, we can always restrict it to a smaller subgroup. Then it may no longer be irreducible — it may break into several connected components.

Fact 22.9 — If we take any V_λ of S_n and restrict it to S_{n-1}, this decomposition is multiplicity-free, so we can write

$$\mathrm{Res}_{S_{n-1}}^{S_n} V_\lambda \cong \bigoplus_{\mu \lessdot \lambda} V_\mu.$$

In particular, this decomposition is unique. Here $\mu \lessdot \lambda$ will turn out to be the covering relation in the Young's lattice, but we don't yet know this. So instead, $\mu \lessdot \lambda$ is the covering relation in the **Bratteli diagram**, a graph that can be defined for our sequence of embeddings — the vertices of the diagram are the elements of $S_n^\wedge$, and the edges are exactly given by the formula above. We will show later that this ends up being Young's lattice.

§22.2 Gelfand–Tsetlin Basis of V_λ

The Gelfand–Tsetliln basis will be a linear basis of V_λ which is unique up to rescaling the basis vectors, defined by the following rule:

Suppose we start with V_λ of S_n, and then we restrict it to S_{n-1}. Then V_λ uniquely breaks into a direct sum of irreducible components, as $\mathrm{Res}^{S_n}_{S_{n-1}} V_\lambda = \bigoplus_{\mu \lessdot \lambda} V_\mu$.

Then we can restrict each V_μ to S_{n-2}, which again breaks it into a direct sum of several components.

We keep on doing this until we arrive to S_1, the trivial group which has only one irreducible representation (the trivial one).

At the end, we have broken our linear space into a direct sum of one-dimensional subspaces. We then take a vector in each component corresponding to representations of S_0.

Example 22.10

For S_3, take the representation

$$V_3 = \{(x_1, x_2, x_3) \in \mathbb{C}^3 \mid x_1 + x_2 + x_3 = 0\}$$

(a 2-dimensional subspace of $\mathbb{C}^3$), where S_3 acts by permutations of x_1, x_2, and x_3.

This is a 2-dimensional irreducible representation of S_3.

The first step is to restrict V_3 from S_3 to S_2. This means we want to only permute x_1 and x_2, and see what possible components we get. We can then write

$$V_3 = \langle(1, 1, -2)\rangle + \langle(1, -1, 0)\rangle,$$

and these are two (invariant) one-dimensional subspaces — we stop here because they're already one-dimensional. These two vectors form the Gelfand–Tsetlin basis. (Sometimes it's convenient to normalize them; we could then divide by $\sqrt{6}$ and $\sqrt{2}$, respectively.)

Example 22.11

For S_4, take V_4 to be the representation

$$V_4 = \{(x_1, x_2, x_3, x_4) \in \mathbb{C}^4 \mid x_1 + x_2 + x_3 + x_4 = 0\},$$

which is again a 3-dimensional subspace of $\mathbb{C}^4$. Construct the Gelfand–Tsetlin basis.

The first step is to restrict from S_4 to S_3. We then have

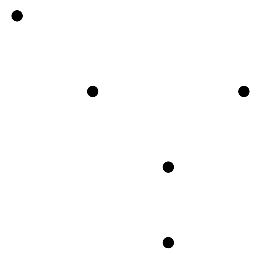
$$\mathrm{Res}^{S_4}_{S_3} V_4 = \{(x_1, x_2, x_3, 0) \mid x_1 + x_2 + x_3 = 0\} \oplus \langle(1, 1, 1, -3)\rangle.$$

So we've broken our three-dimensional subspace as a direct sum of a two-dimensional subspace and a one-dimensional subspace. Then we can restrict the first component from S_3 to S_2. We already know what we get — we then end up with

$$\langle(1, 1, -2, 0)\rangle + \langle(1, -1, 0, 0)\rangle \oplus \langle(1, 1, 1, -3)\rangle.$$

We could again normalize these vectors (by $\sqrt{6}$, $\sqrt{2}$, and $\sqrt{12}$), and we'd get the Gelfand–Tsetlin basis.

Elements of the Gelfand–Tsetlin basis correspond to *paths* in the Bratteli diagram:



When we restrict V_4, we break it into the two elements below it; and so on. So elements of the Gelfand–Tstelin basis for V_λ correspond bijectively to saturated chains in the Bratteli diagram from the minimal element $\hat{0}$ to λ. So we can label our elements of the basis by such chains — we can label the basis elements v_T corresponding to saturated chains T. (These will end up being standard Young tableaux.)

§22.3 Some Facts About Group Algebras

We have a chain of embedded groups, and we can also consider the chain of embedded group *algebras*

$$\mathbb{C}[S_0] \subset \mathbb{C}[S_1] \subset \mathbb{C}[S_2] \subset \cdots.$$

Let Z_n denote the *center* of the group algebra $\mathbb{C}[S_n]$.

Definition 22.12. The **center** of $\mathbb{C}[S_n]$ is the set of elements of the group algebra that commute with all other elements — so we define

$$Z_n = \{f \in \mathbb{C}[S_n] \mid fg = gf \text{ for all } g \in \mathbb{C}[S_n]\}.$$

Understanding the center of our group algebra will be very important for the construction.

An *element* of the group algebra is a formal linear combination of permutations — so

$$f = \sum_{w \in S_n} f_w w,$$

where f_w is some complex number.

Question 22.13. When is $f \in Z_n$?

It's enough to check the case where g is a single permutation — so our condition is that $fu = uf$ for any permutation u. If we express this in terms of our coefficients, then we get that $f_w = f_{uwu^{-1}}$ for all u. So this means $f\colon w \mapsto f_w$ is a constant function on conjugacy classes of S_n — in other words, it only depends on the cyclic type of the permutation.

Definition 22.14. A function on S_n that only depends on the cyclic type is called a **class function** on S_n.

So then Z_n is essentially just the space of class functions on S_n.

§22.4 The Gelfand–Tsteliln Subalgebra

Definition 22.15. The Gelfand–Tsetlin subalgebra of $\mathbb{C}[S_n]$, denoted GT_n, is the sub-algebra generated by Z_1, Z_2, Z_3, $\ldots$, Z_n.

In other words, we have a chain of included group algebras, and for each group algebra we take its center. These centers all live in $\mathbb{C}[S_n]$, and we then take the algebra generated by all these centers.

Lemma 22.16

This is a commutative subalgebra of $\mathbb{C}[S_n]$.

Proof. By definition, Z_n commutes with everything; in particular all elements of Z_n commute with all elements of Z_{n-1}, Z_{n-2}, and so on. So any elements of Z_i and Z_j commute. $\square$

A less trivial lemma is that it's a *maximal* commutative sub-algebra — if we add anything else, we get a noncommutative algebra. Philosophically, in Lie theory there's an important ingredient in the construction of semisimple Lie algebras, called the Cartan subalgebra; this construction is somewhat analogous.

§22.5 (Young)–Jucys–Murphy elements

It turns out we can give an even more explicit construction of this, based on some very important and very simple elements of the group algebra, called **(Young)–Jucys–Murphy elements**.

Remark 22.17. The correct pronounciation of Jucys is with the [j] sound (the person is from Lithuania).

Definition 22.18. The Jucys–Murphy elements are $x_1, \ldots, x_n$ in $\mathbb{C}[S_n]$ defined as:

- $x_1 = 0$.
- $x_2 = (1, 2)$.
- $x_3 = (1, 3) + (2, 3)$.
- In general, $x_i = (1, i) + (2, i) + \cdots + (i - 1, i)$, for all $1 \leq i \leq n$.

These elements have a few useful properties. First,

$$x_i = \sum \text{all transpositions in } S_i - \sum \text{all transpositions in } S_{i-1}.$$

The first term is a class function in S_i, and the second is a class function in S_{i-1}; in other words, the first element belongs to Z_i and the second to Z_{i-1}, and therefore $x_i \in \mathrm{GT}_n$.

The most crucial lemma for the construction is the following:

Theorem 22.19

GT_n is exactly the subalgebra of $\mathbb{C}[S_n]$ generated by the Jucys–Murphy elements $x_1, \ldots, x_n$.

We've already proved half of this theorem — we've shown that all these Jucys–Murphy elements are in the subalgebra. For the converse, we want to show that any element of the group algebra can be expressed in terms of these elements.

All elements of the group algebra are class functions, so we want to show that class functions can be expressed in terms of the Jucys–Murphy elements.

To show this, any class function is a linear combination of

$$c_\lambda = \sum_w w$$

where the sum is over all permutations w of cyclic type λ. So we need to express each of these c_λ in terms of Jucys–Murphy elements.

This can actually be done explicitly by induction; some part of this proof may be left as an exercise in the next problem set. But we will see one simple example:

Example 22.20

The sum of all transpositions $c_{2,1^{n-2}}$ is the sum of all Jucys–Murphy elements.

Exercise 22.21. Try to express the sum of all 3-cycles $c_{31^{n-2}}$ as a certain polynomial in the Jucys–Murphy elements. (Note that the x_i are commutative — x_1x_2 and x_2x_1 are the same thing.)

§23 November 4, 2022

§23.1 Review

Last time, we began discussing the Vershik–Okounkov approach to representations of S_n. We consider the group algebra of S_n, and we consider not just one particular group algebra, but the included sequence of subalgebras

$$\mathbb{C}[S_0] \subset \mathbb{C}[S_1] \subset \mathbb{C}[S_2] \subseteq \cdots.$$

Remark 23.1. The group algebra $\mathbb{C}[S_n]$ has a lot of nontrivial structure; it is a very interesting algebraic and combinatorial object.

In particular, we can look at the *center* of the group algebra,

$$Z_n = Z(\mathbb{C}[S_n]) = \left\{ \sum_{w \in S_n} \mid f_w w \mid f_w \text{ is a class function on } S_n \right\}.$$

More explicitly, f_w has a basis given by the elements c_λ, where c_λ is the sum of all permutations with cycle type λ.

The most important ingredient of this construction is the **Jucys–Murphy elements** $x_1, x_2, \ldots, x_n \in \mathbb{C}[S_n]$ defined as

$$x_i = \sum_{j<i} (j, i).$$

Another important piece is the **Gelfand–Tsetliln subalgebra** $\mathrm{GT}_n \subset \mathbb{C}[S_n]$, the subalgebra generated by Z_1, $\ldots$, Z_n.

Note that all the Jucys–Murphy elements x_i are in GT_n — they're a difference of all the transpositions in S_i and in S_{i-1}.

Theorem 23.2

GT_n is generated by $x_1, \ldots, x_n$.

To prove this, it suffices to prove the following lemma:

Lemma 23.3

Each c_λ can be expressed in $x_1, \ldots, x_n$.

This will be left as an exercise, but we'll look at an example:

Example 23.4

The sum of all transpositions in S_n, $c_{21^{n-2}}$, can be written as $x_1 + \cdots + x_n$.

Example 23.5

The sum of all 3-cycles in S_n, $c_{31^{n-3}}$, can be written in terms of the Jucys–Murphy elements as well.

First we can consider $x_1^2 + \cdots + x_n^2$. We have

$$x_i^2 = \sum_{j_1, j_2 < i} (j_1, i)(j_2, i).$$

Each term here is a 3-cycle if and only if $j_1 \neq j_2$; so we get all three-cycles whose maximal element is i, and we then sum over these three-cycles.

But we have slightly overcounted, because of the possiblity $j_1 = j_2$; in this case we get the identity. So then we have

$$c_{31^{n-3}} = x_1^2 + x_2^2 + \cdots + x_n^2 - \binom{n}{2}.$$

Remark 23.6. This is a good exercise to see the interesting algebraic structure of $\mathbb{C}[S_n]$.

§23.2 The Gelfand–Tsetlin Basis

For each irreducible representation, there is a unique (up to scaling) basis, called the Gelfand–Tstelin basis, constructed by the following rule: we take our irreducible representation of S_n and restrict to S_{n-1}, which breaks it into some representations of S_{n-1}. We then keep doing this, until we get a sum of one-dimensional representations; this gives the basis.

We index our elements of the GT-basis by v_T, where T is a path in the **Bratteli diagram** — in the Bratteli diagram, the vertices are $\bigcup S_n^\wedge$, and the edges correspond to when $\operatorname{Res}_{S_{n-1}}^{S_n}(V)$ of an irreducible representation of S_n contains the corresponding representation in S_{n-1}.

(It will turn out that this is Young's lattice, and T is a standard Young tableau. Right now we will use 'tableau' to refer to paths in the Bratteli diagram.)

Fact 23.7 — $\mathbb{C}[S_n]$ can be identified with the algebra of complex block-diagonal matrices with block sizes d_1, d_2, …, d_N, where $N = |S_n^\wedge|$ is the number of irreducible representations and d_1, …, d_N are their dimensions.

If we want to make this isomorphism explicit, we can pick a linear basis in each irreducible representation. Then each element of the group algebra corresponds to some matrix in this basis, and that's exactly the correspondence.

In particular, we can write down this equivalence with respect to the GT-basis — for each irreducible representation we pick a GT-basis, and then we can explicitly write down the matrix corresponding to an element of the group algebra.

Lemma 23.8

Under this isomorphism, GT_n (the Gelfand–Tsetlin subalgebra) is isomorphic to the subalgebra of all diagonal matrices.

Note that by definition the Gelfand–Tsetlin algebra is commutative, and it's a maximal commutative subalgebra.

This is not very hard to prove using some linear algebra, so we will skip the details; but one fact that might be helpful for the proof is the following:

Proof sketch. First, the center of $\mathrm{Mat}(d \times d)$ is the set of multiples of the identity matrix —

$$Z(\mathrm{Mat}(d \times d)) = \begin{bmatrix} \alpha & & \\ & \alpha & \\ & & \ddots \end{bmatrix}.$$

This means Z_n corresponds to block-diagonal matrices where we have $\alpha, \ldots, \alpha$ on the diagonal in the first block, $\beta, \ldots, \beta$ in the second, and so on.

Then the GT-basis behaves nicely with respect to restrictions; being a bit vague, when we start doing these restrictions the blocks break into smaller blocks, and we end up getting arbitrary diagonal matrices. $\square$

As a corollary, we can now describe the Gelfand–Tsetlin basis in a different way.

Corollary 23.9

The Gelfand–Tsetlin basis $\{v_T\}$ is a unique (up to rescaling) basis of V_λ such that each basis element v_T is a common eigenvector of $x_1, \ldots, x_n$.

This is because we already know the Gelfand–Tsetlin algebra is the algebra generated by the Jucys–Murphy elements, and it should act diagonally on the GT-basis; this means each v_T should be a common eigenvector, and the eigenvalues uniquely determine the v_T — for every v_T there is a sequence of eigenvalues $(\alpha_1, \ldots, \alpha_n)$ such that $x_i \colon v_T \to \alpha_i v_T$.

In particular, this sequence $(\alpha_1, \ldots, \alpha_n)$ uniquely describes the basis element v_T up to rescaling — up to rescaling, there will be a unique eigenvector with these eigenvalues.

Our goal will be to try to understand this collection of eigenvalues somehow. We know that

$$\{\text{tableaux } T\} \leftrightarrow \{(\alpha_1, \ldots, \alpha_n)\}.$$

Definition 23.10. We define the **spectrum** $\mathrm{Spec}(n)$ as the set of all sequences of eigenvalues $(\alpha_1, \ldots, \alpha_n)$ for all v_T in the Gelfand–Tsetlin basis for all irreducible representations V_λ.

Definition 23.11. We define the equivalence relation $\sim$ on $\mathrm{Spec}(n)$ so that $(\alpha_1, \ldots, \alpha_n) \sim (\alpha'_1, \ldots, \alpha'_n)$ if they correspond to basis elements v_T and $v_{T'}$ in the same irreducible representation V_λ.

So we have a certain set of sequences of integers, and some equivalence relation. Our goal is to explicitly and combinatorially describe $\mathrm{Spec}(n)$; somehow we want to see that it's in bijection with standard Young tableaux.

§23.3 Relations on Jucys–Murphy Elements

It turns out that the Jucys–Murphy elements satisfy some nice relations. Our main tool will be the following theorem:

Theorem 23.12

The elements $s_1, \ldots, s_{n-1}$ (adjacent transpositions) and $x_1, \ldots, x_n$ (the Jucys–Murphy elements) of $\mathbb{C}[S_n]$ satisfy the following relations:

- The usual Coxeter relations for $s_1, \ldots, s_{n-1}$.
- $x_i x_j = x_j x_i$ for all i and j.
- $s_i x_j = x_j s_i$ if $j \neq i, i+1$.
- $s_i x_i = x_{i+1} s_i - 1$.
- $s_i x_{i+1} = x_i s_i - 1$.

For the third point, s_i acts on the ith and $(i+1)$th positions; if j is not equal to one of these two special indices, it's easy to see that they commute. So the only nontrivial case is when we have equality, and here we have a slightly different relation.

These relations have a special name — they are called the **degenerate affine Hecke algebra** (DAHA). The Jucys–Murphy elements give an action of the DAHA on the group algebra of S_n.

It turns out that after we check all these relations, we don't have to do any algebra — from these relations, we can deduce all combinatorics of Young tableaux. Somehow, as we will see soon, just from these relations we can formally describe $\operatorname{Spec}(n)$ and see that they correspond to standard Young tableaux.

§23.4 Analyzing the Spectrum

Recall that we have $(\alpha_1, \ldots, \alpha_n) \in \operatorname{Spec}(n)$ if there is some element $v = v_T$ of the Gelfand–Tsetlin basis of V_λ, for some irreducible representation V_λ, such that

$$x_i v = \alpha_i v \text{for all } i = 1, \ldots, n.$$

A priori we just know that the α_i are complex numbers, but we want to describe them explicitly.

First, we know $x_1 = 0$. (This may seem trivial, but it's important — the fact that $x_1 = 0$ tells us something about x_2, because there's a relation involving both x_1 and x_2.) This implies $\alpha_1 = 0$.

The next step is to try to analyze, given α_i, what we can say about α_{i+1}.

Question 23.13. Suppose that $\alpha_i = a$ and $\alpha_{i+1} = b$. What relationships can we find between a and b?

What this means is that $x_i v = av$ and $x_{i+1} v = bv$. We also introduce a vector $v' = s_i(v) \in V_\lambda$. We then have two cases:

Case 1 (v and v' are linearly dependent). We know that $s_i^2 = 1$; so if v' is some multiple of v, the square of the scale factor must be 1. So this means we must have $v' = \pm v$.

Now we want to look at the DAHA, and in particular at the relation $s_i x_i = x_{i+1} s_i - 1$.

Applying x_i to v gives av, and then applying s_i produces av'; so then $s_i x_i(v) = \pm av$. Meanwhile on the right-hand side, $s_i v = v'$, and $x_{i+1} v' = \pm bv'$; this gives

$$\pm av + v = \pm bv$$

(where the signs are consistent). This means

$$b = a \pm 1.$$

Case 2 (v and v' are linearly independent). Then $\operatorname{Span}(v, v')$ is a two-dimensional subspace $\langle v, v' \rangle \subset V_\lambda$.

Claim — On this two-dimensional subspace, x_{i+1}, x_i, and s_i act as

$$x_i = \begin{bmatrix} a & -1 \\ 0 & b \end{bmatrix}, \; x_{i+1} = \begin{bmatrix} b & 1 \\ 0 & a \end{bmatrix}, \; s_i = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

The first statement means that $x_i v = av$ (from the first column) and $x_i v' = -v + bv'$ (from the second column). This again follows from the relations.

We'll finish the proof next lecture, but right now we'll discuss what we are going to get. Somehow, from the DAHA we can describe properties of the spectrum:

Theorem 23.14

If $(\alpha_1, \dots, \alpha_n) \in \operatorname{Spec}(n)$, then:

(1) $\alpha_i = 0$.

(2) $\alpha_i \neq \alpha_{i+1}$.

(3) If $\alpha_i = \alpha_{i+1} \pm 1$, then $s_i(v_T) = \pm v_T$ — the ith transposition sends our basis element to $\pm$ of itself.

(4) If $\alpha_i \neq \alpha_{i+1} \pm 1$, then the element $\widetilde{\alpha}$ obtained by switching α_i and α_{i+1} also belongs to $\operatorname{Spec}(n)$.

(5) We cannot have $\alpha = (\dots, a, \dots, a+1, \dots, a) \in \operatorname{Spec}(n)$.

These are the properties we'll get — they're easy to show by algebraic manipulations — and in fact, this *uniquely* determines the spectrum, and we can deduce that this object is in bijection with standard Young tableaux. The main point is that if two entries differ by something other than 1, we can switch them and the result will also belong to the spectrum.

§24 November 7, 2022

Recall that V_λ denotes the irreducible representations of S_n, labeled by $\lambda \in S_n^\wedge$ (which we'll currently think of as a formal set of labels for the irreducible representations — we are assuming that we so far know nothing about Young diagrams and Young tableaux).

An important ingredient is the Jucys–Murphy elements

$$x_i = \sum_{j<i} (j, i) \in \mathbb{C}[S_n].$$

Last time, we discussed that each irreducible representation V_λ has a unique (up to rescaling) basis, called the GT–basis; we could define the GT–basis by restricting to S_{n-1} and breaking into pieces, but we could also say that this basis $\{v_T\}$ is given by the common vectors of $x_1, \dots, x_n$.

So then we want to analyze the possible common eigenvectors of the Jucys–Murphy elements.

Moreover, for any $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{C}^n$, there's at most one basis element such that $x_i \colon v_T \to \alpha_i v_T$. This means we can label v_T by these sequences α, instead of by paths in the Bratteli diagram.

Remark 24.1. One thing we haven't proven is that if we restrict our irreducible representation to S_{n-1}, all multiplicities are 0 or 1; that's a key fact why the construction works, that we have not proven. But it can be proven by checking some properties of the group algebra.

Definition 24.2. We use $\operatorname{Spec}(n)$ to denote the set of all $\alpha = (\alpha_1, \dots, \alpha_n)$ for all v_T for all V_λ, with the equivalence relation $\alpha \sim \beta$ if α and β correspond to vectors in the same V_λ.

A priori this is some set of complex vectors α; we'd like to understand it.

We can check explicitly from the definitions that the Jucys–Murphy elements satisfy the DAHA relations. Now we want to perform some manipulations and somehow get a description of α.

§24.1 Analysis of Spec(n)

First, we know that $\alpha_1 = 0$ (since $x_1 = 0$).

Now let's consider two consecutive entries $\alpha_i = a$ and $\alpha_{i+1} = b$; and let $v = v_T$ be the vector corresponding to α. Then we must have

$$x_i \colon v \to av \text{ and } x_{i+1} \colon v \to bv.$$

We also define $v' = s_i(v)$. Now we consider two cases:

Case 1 (v and v' are linearly dependent). Then since $s_i^2 = 1$, the only possible linear dependence is $v' = \pm v$. Now from the DAHA relations, namel from $s_i x_i = x_{i+1} s_i - 1$, we have

$$\pm av = \pm bv - v.$$

This means $b = a \pm 1$. (More specifically, if we have $v' = +v$ then $b = a+1$, while if $v' = -v$ then $b = a-1$.)

Case 2 (v and v' are linearly independent). In this case, we consider the two-dimensional subspace $\langle v, v' \rangle \subseteq V_\lambda$ spanned by these vectors. We can explicitly write how x_i and x_{i+1} act on this subspace — we know

$$x_i = \begin{bmatrix} a & -1 \\ 0 & b \end{bmatrix} \text{ and } x_{i+1} = \begin{bmatrix} b & 1 \\ 0 & a \end{bmatrix}.$$

What this means is that $x_i \colon v \mapsto av$ (this is true by definition), and

$$x_i \colon v' \mapsto x_i(s_i v) = (s_i x_{i+1} - 1)v = -v + bv'$$

(using the final DAHA relation). We can do a similar calculation for x_{i+1}. In particular, we can deduce some corollaries from this:

Lemma 24.3

We have $a \neq b$.

Proof. Otherwise, we would have

$$x_i = \begin{bmatrix} a & -1 \\ 0 & a \end{bmatrix}.$$

This gives us a nontrivial Jordan block in x_i. But we know that the matrix x_i should be diagonalizable — because in the GT–basis, it acts diagonally. (We know x_i is given by a *big* matrix over all of V_λ, and that big matrix must be diagonalizable; and we're looking at this small 2-dimensional subspace. But any matrix for which there exists a 2-dimensional subspace with $a = b$ would not be diagonalizable — if it has a nontrivial Jordan block with respect to some subspace, then it cannot have a trivial Jordan decomposition.) So this is a contradiction. □

So α cannot have two repeated entries.

Now let's introduce another vector

$$\widetilde{v} = v + (a - b)v',$$

which is another vector living in our two-dimensional subspace — in fact $\langle v, v' \rangle = \langle v, \widetilde{v} \rangle$. We will use this basis for our two-dimensional piece instead.

The reason we like $\widetilde{v}$ better is that by again applying these relations, we can see

$$x_i \colon \widetilde{v} \mapsto x_i(v + (a-b)v') = av + (a-b)(-v + bv') = b(v + (a-b)v') = b\widetilde{v}$$

(using our calculation from earlier). Similarly, we can check that

$$x_{i+1} \colon \widetilde{v} \mapsto a\widetilde{v}.$$

More generally, $x_j \colon \widetilde{v} \mapsto \alpha_j \widetilde{v}$ for all $j \neq i, i+1$.

In particular, $\widetilde{v}$ is also a common eigenvector of all $x_1, \ldots, x_n$, so up to scaling, it must also be an element of the Gelfand–Tsetlin basis. Furthermore, its vector of eigenvalues $\widetilde{\alpha}$ is given by

$$\widetilde{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{i+1}, \alpha_i, \ldots, \alpha_n)$$

(where we only swap the ith and $(i+1)$st entry). In particular, $\widetilde{\alpha}$ is also an element of $\operatorname{Spec}(n)$.

Lemma 24.4

In the second case, $a - b \neq \pm 1$.

Proof. This lemma is quite tricky. Otherwise, assume that $a - b = \pm 1$. Then $\widetilde{v} = v \pm s_i(v)$. Now applying s_i, we get

$$s_i \widetilde{v} = s_i(v) \pm v = \pm \widetilde{v}.$$

This means $\widetilde{v}$ lives in Case 1 — it's also an element of the Gelfand–Tsetlin basis, and we're in the situation where s_i sends it to a multiple of itself. So by the first case, we deduce that $b = a \pm 1$. But this means $a - b = \pm 1$. This is the opposite sign as we assumed (that's the trickiest part of the argument), so this is a contradiction. $\square$

Student Question. *Is $\widetilde{\alpha}$ only in Spec(n) if we're in Case 2?*

Answer. Yes.

In the general case, we have the following:

Lemma 24.5

Suppose that $(\alpha_1, \ldots, \alpha_i, \alpha_{i+1}, \ldots, \alpha_n) \in \operatorname{Spec}(n)$. Then:

(1) $\alpha_{i+1} \neq \alpha_i$.

(2) There are two possibilities:

- $\alpha_i = \alpha_{i+1} \pm 1$ if and only if we are in Case I.
- $\alpha_i \neq \alpha_{i+1} \pm 1$ if and only if we are in Case II.

Furthermore, in Case II we can swap the two entries — $(\alpha_1, \ldots, \alpha_{i+1}, \alpha_i, \ldots, \alpha_n)$ is also in $\operatorname{Spec}(n)$.

This will be a crucial observation — we can swap two entries only if their difference is not 1.

We'll now see a few easy lemmas.

Lemma 24.6

We cannot have $(\alpha_1, \ldots, \alpha_n) = (\ldots, a, a \pm 1, a, \ldots)$.

Proof. Assume for contradiction that we have this situation; for simplicity assume we have $(\ldots, a, a + 1, a, \ldots)$. Now we apply $s_i s_{i+1} s_i$ to the corresponding vector v. So far, because the first difference is 1 we are in case I, so $s_i(v) = v$, so then

$$s_i s_{i+1} s_i(v) = s_i s_{i+1}(v).$$

But then when we apply s_{i+1}, we're also in Case I, but with -1; so then $s_{i+1}(v) = -v$. Then we again have the same vector and want to apply s_i again; since the difference is 1, we have $s_i(-v) = -v$. So we have changed sign once.

But this also equals

$$s_{i+1} s_i s_{i+1}(v).$$

By using the same argument, every time we apply s_{i+1} we change the sign, and every time we apply s_i we don't change the sign; so here we'd be changing the sign twice, and we'd get v.

But $v \neq -v$, so this is a contradiction. (The same proof applies if we have $a - 1$ instead; then we get $+v$ in the first case and $-v$ in the second.) $\square$

Let's now summarize everything that we've proven so far:

Definition 24.7. An **allowed transposition** is a transposition $(\alpha_1, \ldots, \alpha_n) \mapsto (\alpha_1, \ldots, \alpha_{i+1}, \alpha_i, \ldots, \alpha_n)$ if $\alpha_{i+1} \neq \alpha_i \pm 1$.

In other words, we can swap two consecutive entries if their difference is not ± 1. Then we can summarize what we've proved as follows:

Theorem 24.8

For any $\alpha = (\alpha_1, \ldots, \alpha_n) \in \text{Spec}(n)$, we have:

- $\alpha_1 = 0$.
- $\alpha_{i+1} \neq \alpha_i$ for any i.
- We cannot have a pattern of $(\ldots, a, a \pm 1, a, \ldots)$.
- For any allowed transposition $\beta = (\alpha_1, \ldots, \alpha_{i+1}, \alpha_i, \ldots, \alpha_n)$, we also have $\beta \in \text{Spec}(n)$, and $\beta \sim \alpha$.

It turns out that these properties are all that we need to know in order to combinatorially characterize our set.

From these properties, we will write down a few corollaries:

Corollary 24.9

If we have $(\alpha_1, \ldots, \alpha_n) \in \text{Spec}(n)$, then $\alpha_i \in \mathbb{Z}$ for all i.

Proof. Suppose one were not an integer. Then find the first non-integer entry. We can perform an allowed transposition with everything before it — they're all integers, so the differences cannot be ± 1. This means we can move this non-integer entry to the first position. But we know that in the first position we should have 0, which is a contradiction. $\square$

Corollary 24.10

If $\alpha_i = \alpha_j = a$ with $i < j$, then both $a - 1$ and $a + 1$ appear in $\{\alpha_{i+1}, \ldots, \alpha_{j-1}\}$.

We can't have repeated entries immediately one after the other, but we can have repeated entries some distance apart.

Proof. We again use contradiction. Suppose not, so that we've found two entries a and a, and either $a + 1$ or $a - 1$ does not appear between them. Find such a bad pair (i, j) with minimal $j - i$.

Let's now look at the entry immediately after the ith position. This entry must equal $a \pm 1$ — otherwise we could commute these two entries to decrease the distance. Similarly, the entry right before the jth position must also equal $a \pm 1$. If we have opposite signs, that's what we wanted to prove. So the only situation is when both are of the same sign, in which case we have a bad pair with smaller distance — since a does not appear in between them. □

Corollary 24.11

For all $i > 1$, we have that either $\alpha_i + 1$ or $\alpha_i - 1$ belongs to $\{\alpha_1, \ldots, \alpha_{i-1}\}$.

Proof. There are two cases to consider.

Case 1 ($\alpha_i = 0$). Then by the property we just proved, between the two 0's we should have both $+1$ and -1; so *both* must appear before the ith entry.

Case 2 ($\alpha_i \neq 0$). Then if this weren't true, all preceding entries wouldn't be $\alpha_i + 1$ or $\alpha_i - 1$, so by using allowed transpositions we could move α_i all the way to the first position. This would produce a nonzero entry in the first position, contradiction. □

We claim that these properties are all that we need to characterise $\mathrm{Spec}(n)$. We are out of time, so we will do this next lecture. But if we have nothing to do before Wednesday, we can figure it out ourselves:

We want to represent our α_i by an *abacus*, where we have strings labelled $\ldots, -2, -1, 0, 1, 2, \ldots$. These beads correspond to α_i, and the string positions to their values.

§25 November 9, 2022

Recall that V_λ are the irreducible representations of S_n, where λ are currently formal indices — we want to see that they correspond to Young diagrams. Each V_λ has a Gelfand–Tsetlin basis $\{v_T\}$, and we will also see that this basis is labelled naturally by standard Young tableaux.

Earlier we saw that we could label basis vectors by the vector of eigenvalues — $v_t \leftrightarrow \alpha = (\alpha_1, \ldots, \alpha_i)$, where α_i is the eigenvalue of v_T in the ith Jucys–Murphy element — so $x_i \colon v_T \mapsto \alpha_i v_T$. We denote

$$\mathrm{Spec}(n) = \{(\alpha_1, \ldots, \alpha_n)\}$$

over all basis elements v_T of all irreducible representations of S_n, and we write $\alpha \sim \beta$ if α and β correspond to basis elements of the same irreducible representation.

Last class, we used the DAHA relations to show that $\mathrm{Spec}(n)$ satisfies a bunch of conditions. Our goal is to describe $\mathrm{Spec}(n)$ and the equivalence relation combinatorially.

Theorem 25.1

For any $\alpha \in \mathrm{Spec}(n)$:

(1) $\alpha_1 = 0$;

(2) For all $i = 1, \ldots, n$, either $\alpha_i - 1$ or $\alpha_i + 1$ occurs in $\{\alpha_1, \alpha_2, \ldots, \alpha_{i-1}\}$.

(3) If $\alpha_i = \alpha_j = a$ with $i < j$, then both $a - 1$ and $a + 1$ appear in $\{\alpha_{i+1}, \ldots, \alpha_{j-1}\}$.

(4) Any β obtained from α by allowed transpositions belongs to $\mathrm{Spec}(n)$, and furthermore $\alpha \sim \beta$.

Definition 25.2. An **allowed transposition** is an operation $(\alpha_1, \ldots, \alpha_n) \mapsto (\alpha_1, \ldots, \alpha_{i+1}, \alpha_i, \ldots, \alpha_n)$ if $\alpha_{i+1} \neq \alpha_i \pm 1$ (we can swap two adjacent entries if their values aren't adjacent).

We now have a list of necessary conditions on $\mathrm{Spec}(n)$; but as we will see soon, these conditions uniquely characterize $\mathrm{Spec}(n)$. In particular note that (1) and (2) together imply that all α_i are integers.

Definition 25.3. Let $\mathrm{Cont}(n) \subseteq \mathbb{Z}^n$ be the set of integer vectors $\alpha = (\alpha_1, \ldots, \alpha_n)$ such that α satisfies (1) – (3), and any allowed transposition (or sequence of such transpositions) of α also satisfies (1) – (3).

Notation 25.4. We use $\approx$ to denote the equivalence relation on $\mathrm{Cont}(n)$ generated by allowed transpositions.

By the theorem, we know that $\mathrm{Spec}(n) \subseteq \mathrm{Cont}(n)$, and moreover that if $\alpha \approx \beta$ (α and β are obtained from each other by a sequence of allowed transpositions), then $\alpha \sim \beta$.

We want to show that all of these are equalities.

To show this, we know that the number of equivalence classes in $\mathrm{Spec}(n)$ is at most the number of equivalence classes in $\mathrm{Cont}(n)$ — so

$$\#\mathrm{Spec}(n)/\!\sim \leq \#\mathrm{Cont}(n)/\!\approx .$$

But the left-hand side is the number of irreducible representations of S_n, and therefore of conjugacy classes — and therefore the left-hand side is $p(n)$. So we'd like to show that the right-hand side also is $p(n)$.

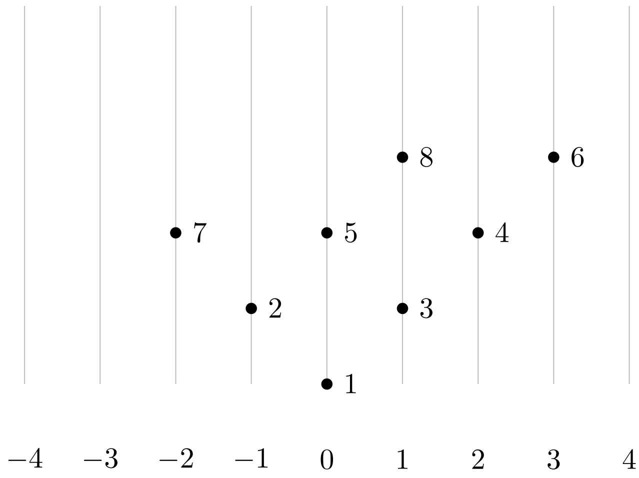
Example 25.5

Some examples of $\mathrm{Cont}(n)$:

n	$\mathrm{Cont}(n)$
1	(0)
2	$(0,1)$, $(0,-1)$
3	$(0,1,2)$, $(0,1,-1)$, $(0,-1,-2)$, $(0,-1,1)$

In the case for $n = 3$, we see only one allowed transposition — so $(0,1,-1) \approx (0,-1,1)$.

We could keep going and get a list of all vectors we can get, but it'll help to represent $(\alpha_1, \ldots, \alpha_n) \in \mathrm{Cont}(n)$ by an *abacus*, with n beads labelled $1, 2, \ldots, n$.



We say that a bead can move past another bead if they're not consecutive. Every time we drop a bead, it must bump into something. Meanwhile, when we drop 5 onto the same string as 1, we must have something on both sides.

This particular abacus represents $(0, -1, 1, 2, 0, 3, -2, 1)$. In general, the ith bead is on the α_ith string.

This picture should remind us of something — it looks like a Young diagram in a different orientation. And our beads are labelled by numbers; they will actually give us a standard Young tableau.

Theorem 25.6

$\mathrm{Cont}(n)$ is in bijection with the set of all SYT with n boxes, and $\alpha \approx \beta$ if and only if they correspond to SYT of the same shape.

This bijection is just given by rotation and taking the mirror image (so that 0 is the upper-left corner, and the diagonal on the right is the first row) — for example,

$$\alpha = (0, -1, 1, 2, 0, 3, -2, 1) \rightsquigarrow \begin{array}{|c|c|c|c|} \hline 1 & 3 & 4 & 6 \\ \hline 2 & 5 & 8 & \\ \cline{1-3} 7 & & & \\ \cline{1-1} \end{array}.$$

In terms of standard Young tableaux, an allowed transposition corresponds to switching α_i and α_{i+1} if their difference is not ± 1; in the picture, it means we switch the bead with label i and $i+1$ if they're not on adjacent strings. So we are switching the boxes with i and $i+1$, and we can do this if and only if they're not in adjacent diagonals. (If i is in the ath diagonal, and $i+1$ in the bth diagonal, then we can swap them if $a - b \neq \pm 1$.)

If i and $i+1$ were in the same diagonal then we *couldn't* switch them, since we'd have a horizontal or vertical domino; otherwise we can.

We're not going to prove this, since it's fairly straightforward and the picture should convince us. But one ingredient of the proof is the following:

Lemma 25.7

Any two standard Young tableaux of the same shape can be obtained from each other by a sequence of allowed transpositions.

(This will be left to us as an exercise; it is the only nontrivial ingredient in the proof of this theorem.)

Now we are basically done — we know that $\mathrm{Cont}(n)$ is in bijection with all standard Young tableaux, and its equivalence classes correspond to Young diagrams with n boxes. In particular, $\#\mathrm{Cont}(n)/{\approx} = p(n)$ as well. This implies that $\mathrm{Spec}(n) = \mathrm{Cont}(n)$ and $\sim = \approx$.

So we have constructed an explicit basis for our irreducible representations labelled by standard Young tableaux, and so on. Moreover, from this construction we can pretty easily see that now the Bratteli diagram is just Young's lattice — previously we knew that it was just some graph, but now we know that its nodes correspond to Young diagrams.

Corollary 25.8

The Bratteli diagram for $S_0 \subset S_1 \subset S_2 \subset \cdots$ is Young's lattice, and the restrictions can be written as

$$\mathrm{Res}^{S_n}_{S_{n-1}} V_\lambda = \bigoplus_{\mu \lessdot \lambda} V_\mu,$$

where the sum is over all μ obtained from λ by removing a single box.

To see this, we'd like to describe $\mathrm{Res}^{S_{n-1}}_{S_n}$ in terms of our vectors $(\alpha_1, \ldots, \alpha_n)$. When we restrict to S_n, we have the same $\alpha_1, \ldots, \alpha_{n-1}$ and just forget about α_n.

Remark 25.9. In this correspondence, a Young tableau corresponds to $(\alpha_1, \ldots, \alpha_n)$ where α_i equals the **content** of the entry i in T. (Recall that the content of a box is $j - i$, where j is its column and i its row.)

When we remove α_n, we're simply deleting the box containing the maximal entry.

So we get all the usual combinatorics from this construction. But we actually get more, because we can explicitly describe the action of the symmetric group on the basis elements.

§25.1 Young's Orthogonal Form

Question 25.10. What is the action of S_n on the Gelfand–Tsetlin basis?

So for every element of S_n, we want to find some matrix whose rows and columns are labelled by Young tableaux.

It's enough to describe the action of the *generators* $s_1, \ldots, s_{n-1}$ (the adjacent transpositions) — then for any permutation, we can take a product of matrices to get its action.

Theorem 25.11 (Young's Orthogonal Form)

Fix $\lambda \vdash n$, and consider the Gelfand–Tsetlin basis $\{v_T \mid T \in \mathrm{SYT}(\lambda)\}$. Then there are two cases for $s_i v_T$:

(1) $s_i\colon v_T \mapsto \pm v_T$, if the two entries i and $i+1$ are in adjacent diagonals (if they form a horizontal domino then $s_i\colon v_T \mapsto v_T$, and if they form a vertical domino then $s_i\colon v_T \mapsto -v_T$).

(2) If i and $i+1$ are *not* in adjacent diagonals, then suppose that i is in diagonal $\alpha_i = a$, and $i+1$ is in the diagonal $\alpha_{i+1} = b$, with $a - b \neq \pm 1$. Then let $\widetilde{T}$ be the tableau obtained from T by switching i and $i+1$ (an allowed transposition). Then

$$s_i\colon v_T \mapsto \frac{1}{b-a} \cdot v_T + \sqrt{1 - \frac{1}{(b-a)^2}}\, v_{\widetilde{T}}.$$

The first case is exactly the situation we saw earlier — the horizontal domino case is when $\alpha_{i+1} = \alpha_i + 1$, and the vertical case is when $\alpha_{i+1} = \alpha_i - 1$.

The second formula follows from the second case of the previous lecture. We had v and $s_i v = v'$, and then we had $\widetilde{v}$, which was an element of the GT–basis; up to rescaling we had $\widetilde{v} = v_T$.

In the previous class we didn't have any square roots; but the GT–basis is only defined up to scaling, so here we're rescaling the vectors so that they all have length 1. If we take the formulas from the previous lectures —

$$\widetilde{v} = v_T + (a - b)s_i(v)$$

times some normalizing constant — then we can deduce this formula.

There is one slight issue we are going to suppress here — these vectors are only defined up to switching sign. So the sign could be + or −; but it turns out that having all + is an allowed choice. (These details will be left to us to figure out.)

Example 25.12

Take $\lambda = \yng(2,1)$, so we have two basis elements

$$v_{\young(12,3)} \text{ and } v_{\young(13,2)}.$$

Then we have

$$s_1 \colon v_{\young(12,3)} \mapsto +v_{\young(12,3)}$$

(since 1 and 2 are in a horizontal domino), and

$$v_{\young(13,2)} \mapsto -v_{\young(13,2)}.$$

So as a matrix in this basis, s_1 becomes the diagonal 2×2 matrix

$$s_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Now let's consider s_2, which is slightly more complicated. Boxes 2 and 3 are not in adjacent diagonals, so we get

$$s_2 \colon v_{\young(12,3)} \mapsto -\frac{1}{2} \cdot v_{\young(12,3)} + \frac{\sqrt{3}}{2} v_{\young(13,2)}$$

and similarliy

$$s_2 \colon v_{\young(13,2)} \mapsto \frac{1}{2} v_{\young(13,2)} + \frac{\sqrt{3}}{2} v_{\young(12,3)}.$$

As a matrix, this means

$$s_2 = \begin{bmatrix} -1/2 & \sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{bmatrix}.$$

Note that these are both reflections: if we draw the first vector horizontally and the second vertically, then s_1 is a reflection across the horizontal line, and s_2 is a reflection across another line forming a $\pi/3$ angle with the horizontal.

§26 November 14, 2022

§26.1 Young's Orthogonal Form

There's an explicit construction for V_λ of S_n — as a linear space, $V_\lambda \cong \mathbb{C}^{f^\lambda}$, and it has an explicit basis $\{v_T \mid T \in \mathrm{SYT}(\lambda)\}$. It is convenient to represent SYT by their **content vectors** $c(T) = (c_1, \ldots, c_n)$, where c_i is the **content** of the box containing i in T. (Note that the *content* of a box is not its entry — the content is the diagonal on which the entry is located.)

Example 26.1

One example of the contents of a SYT is

$$T = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 5 \\ \hline 4 \\ \cline{1-1}\end{array} \rightsquigarrow c(T) = (0, 1, -1, -2, 0).$$

In the Vershik–Okounkov construction, we denoted the contents by $\alpha_1, \ldots, \alpha_n$ because they were eigenvalues of the Jucys–Murphy elements; today we will use c_i to denote that they are contents of boxes.

Then we can explicitly tell how the generators of the symmetric group act on the basis elements: they act by the matrices

$$R_{s_i} \colon v_T \mapsto \begin{cases} v_T & \text{if } c_{i+1} - c_i = 1 \\ -v_T & \text{if } c_{i+1} - c_i = -1 \\ \frac{1}{c_{i+1}-c_i} v_T + \sqrt{1 - \frac{1}{(c_{i+1}-c_i)^2}} v_{\tilde{T}}, \end{cases}$$

where $\tilde{T}$ is the tableaux obtained from swapping i and $i+1$ in $\tilde{T}$. (The first two cases correspond to case 1 from earlier, where we have a horizontal or vertical domino; the third corresponds to case 2, where we can switch i and $i+1$.)

Remark 26.2. The last expression is true in all cases, since in the first two the second terms is 0 — but $\tilde{T}$ is not really defined there.

We can essentially derive this directly from Vershik–Okounkov; the only thing that isn't entirely obvious is that we can make all the signs in the square roots positive. We can immediately check that these matrices R_{s_i} satisfy the Coxeter relations, and that the eigenvalues of the Jucys–Murphy elements are exactly the c_i; this means it must agree with the Vershik–Okounkov construction.

Example 26.3

When $n = 3$ and
$$\lambda = \begin{array}{|c|c|}\hline & \\ \hline & \multicolumn{1}{c}{} \\ \cline{1-1}\end{array},$$
we have two basis elements v_1 and v_2 corresponding to $\begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1}\end{array}$ and $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1}\end{array}$ respectively. We can check that R_{s_1} sends $v_1 \mapsto v_1$ and $v_2 \mapsto -v_2$, so
$$R_{s_1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$
Meanwhile, R_{s_2} maps
$$v_1 \mapsto \frac{1}{-2}v_1 + \frac{\sqrt{3}}{2}v_2 \text{ and } v_2 \mapsto \frac{1}{2}v_2 + \frac{\sqrt{3}}{2}v_1,$$
giving the matrix
$$R_{s_2} = \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix}.$$
So then if we draw v_1 and v_2 as orthogonal vectors, then s_1 is a reflection with respect to the x-axis, and s_2 is a reflection over the line with an angle of $\pi/3$. If we look at the group generated by these reflections, then we also get a reflection over the line with angle $2\pi/3$ — from $s_1 s_2 s_1$.

We can see that this is the action of S_3 on $\{(x_1, x_2, x_3) \mid x_1 + x_2 + x_3 = 0\}$ by permutation of coordinates. (These three lines form a *Coxeter arrangement*.)

Exercise 26.4. This generalizes to S_n — if
$$\lambda = (n-1, 1) = \begin{array}{|c|c|c|c|c|}\hline & & & & \\ \hline & \multicolumn{4}{c}{} \\ \cline{1-1}\end{array},$$
then V_λ is isomorphic to the action of S_n by permutation of coordinates on the space $\{(x_1, \dots, x_n) \in \mathbb{C}^n \mid x_1 + \cdots + x_n = 0\}$.

§26.2 Characters of Representations of S_n

Definition 26.5. Given a representation $R\colon S_n \to \mathrm{GL}(V)$, then its **character** χ_V is the function $\chi_V\colon S_n \to \mathbb{C}$ sending $w \mapsto \mathrm{tr}(R(w))$ for each $w \in S_n$.

Recall that the *trace* is the sum of diagonal entries. It has the important property $\mathrm{tr}(AB) = \mathrm{tr}(BA)$ — or equivalently, $\mathrm{tr}(A) = \mathrm{tr}(CAC^{-1})$. So the trace doesn't change when we conjugate matrices.

This gives a few important properties about characters.

Proposition 26.6

General facts about characters:

- χ_V does not depend on the choice of basis in V.
- χ_V is a class function — it is constant on conjugacy classes of S_n.
- Two representations V_1 and V_2 are isomorphic if and only if $\chi_{V_1} = \chi_{V_2}$.
- The character of $V_1 \oplus V_2$ is $\chi_{V_1} + \chi_{V_2}$.
- The character of the tensor product $V_1 \otimes V_2$ is $\chi_{V_1} \cdot \chi_{V_2}$.

These facts apply to any finite group. In particular, if we want to find the character of V_λ, we can pick any basis, so we may as well pick the Gelfand–Tsetlin basis.

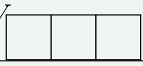
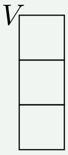
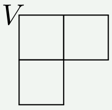
The second fact means that the value of the character only depends on the *cycle type*.

The fourth point is because the trace of a block-diagonal matrix is the sum of traces in the two blocks.

These properties show that characters are important ways to describe the representation; in representation theory, it's important to calculate the character of irreducible representations.

Example 26.7

The character table for S_3 is as follows (with the columns describing conjugacy classes):

	Id	(12)	(123)
V 	1	1	1
V 	1	-1	1
V 	2	0	-1

In general, this table has dimensions $p(n) \times p(n)$.

To find the first row, every matrix acts by 1 (this is the trivial representation), and the trace is always 1. The second is the *sign* representation, where every element acts by the 1×1 matrix $\operatorname{sgn}(w)$.

For the third row, the identity element always acts by the identity matrix; so the trace of the identity matrix is the size of the matrix, and we know this is a 2-dimensional representation. To fill in the rest, recall that

$$R_{s_1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

which has trace 0. To find the last entry, we have

$$R_{s_2} = \begin{bmatrix} -1/2 & \sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{bmatrix},$$

and we can calculate $\operatorname{tr}(R_{s_1} R_{s_2}) = -1$.

Remark 26.8. The characters have additional nice orthogonality properties that can be used to quickly obtain the last entry.

We would like to make a similar table for any symmetric group S_n.

Definition 26.9. We define $\chi_\lambda(\mu)$ as the values of $\chi_{V_\lambda}(w)$ for permutations w of cyclic type μ.

Theorem 26.10 (Murnaghan–Nakayama Rule)
We have

$$\chi_\lambda(\mu) = \sum_T (-1)^{\mathrm{ht}(T)}$$

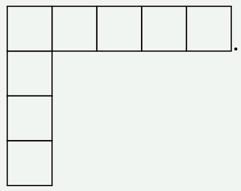
where the sum is over all **ribbon tableaus** T of shape λ and type μ.

Of course, we need to define all of these terms.

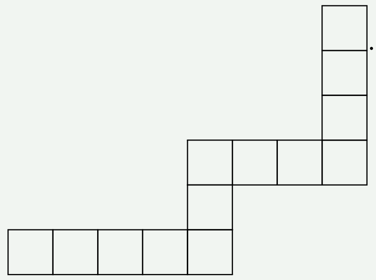
Definition 26.11. A **ribbon** is a skew shape φ/ψ which is connected and contains no 2×2 box.

Example 26.12

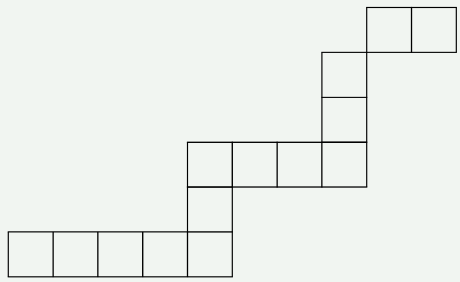
A hook is a ribbon:



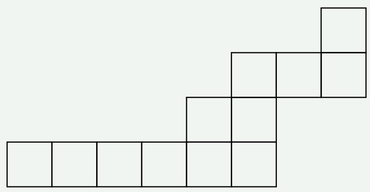
More generally, we could also have



On the other hand,



is not a ribbon, because it is disconnected;



is not a ribbon because it has a 2×2 box.

Remark 26.13. The ribbon does not need to go all the way to the top row or left column.

Definition 26.14. A k-ribbon contains k boxes.

Definition 26.15. A **ribbon tableau** of shape λ and type $\mu = (\mu_1, \ldots, \mu_k)$ is a reverse plane partition of shape λ, such that all boxes with entries i form a μ_i-ribbon.

Definition 26.16. A **reverse plane partition** is a filling of the shape with numbers $1, \ldots, k$ that weakly increase in rows and columns.

Remark 26.17. We saw a problem about domino tableaux on the problem set; domino tableaux are ribbon tableaux of type $(2, 2, 2, \ldots)$.

Definition 26.18. The **height** of a ribbon is ht(ribbon) = #rows − 1.

Example 26.19

The height of a single row would be 0:

$$\mathrm{ht}\left(\ydiagram{5}\right) = 0.$$

Meanwhile,

$$\mathrm{ht}\left(\ydiagram{3+5,4}\right) = 1.$$

Definition 26.20. The **height** of a tableaux T is

$$\mathrm{ht}(T) = \sum_i \mathrm{ht}(i\text{th ribbon}).$$

Example 26.21

Suppose that

$$T = \begin{ytableau} 1 & 1 & 2 & 4 \\ 1 & 2 & 2 & 4 \\ 3 & 3 & 4 & 4 \\ 3 & 4 & 4 \\ 3 & 5 & 5 \end{ytableau}.$$

Then the shape is $\lambda = (4, 4, 4, 3, 3)$, and the type $\mu = (3, 3, 4, 6, 2)$. We then have

$$\mathrm{ht}(T) = 1 + 1 + 2 + 3 + 0 = 7.$$

Remark 26.22. Note that here μ is not actually a partition; this is possible. In fact, the theorem is still true even if μ is not a partition — it gives a rule for $\chi_\lambda(\mu)$ given by any ordering of the parts of μ.

Example 26.23

Find $\chi_{(2,1)}((3, 0))$.

This means we want the bottom-right entry of the character table for S_3. Here our shape is $(2, 1)$, and we want to fill it with a single ribbon. There is only one ribbon tableaux $\begin{ytableau} 1 & 1 \\ 1 \end{ytableau}$, with height 1; so we get -1.

Example 26.24

Find $\chi_{(2,2)}(2, 2)$.

Then we want domino tableaux of shape $\begin{array}{|c|c|}\hline & \\ \hline & \\ \hline\end{array}$. There are two:

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

The first gives $(-1)^0$ and the second gives $(-1)^2$, so we get an answer of 2.

Student Question. *Why does the height appear but the width doesn't?*

Answer. As we can see, this rule is not symmetric — the rows and columns play different roles. But there should be some symmetry, which we will discuss later; there are some nontrivial identities involving ribbon tableaux, which may appear in the next problem set.

We will formulate a more general version of the Murnaghan–Nakayama rule — it is true not only for usual Young diagrams, but also for *skew* Young diagrams.

Definition 26.25. For a skew shape λ/μ with n boxes, we can define the representation $V_{\lambda/\mu}$ of S_n by Young's orthogonal form acting on $\{v_T\}$, with the basis given by SYT of skew shape (by the same formulas).

This is not necessarily an irreducible representation, but it is still a representation, and we can still define its character in the same way.

The Murnaghan–Nakayama rule also works in this more general situation:

Theorem 26.26

We have

$$\chi_{\lambda/\mu}(\nu) = \sum_T (-1)^{\mathrm{ht}(T)},$$

where λ/μ is a skew shape and $\nu = (\nu_1, \dots, \nu_k)$ is any composition of n (with parts not necessarily in decreasing order).

In particular, this gives many different rules (one for each rearrangement of parts of ν.)

It'll be easier for us to prove this generalization to skew shapes.

The idea is that we have an explicit action given by Young's orthogonal form, and that means we can explicitly calculate the trace of the product of matrices. But there are some choices we can make, that simplify our lives. We have operators R_{s_i} acting on our basis elements by $v_T \mapsto \frac{1}{c_{i+1}-c_i} v_T + (\cdots) v_{\widetilde{T}}$. Then for every permutation we look at the product of these matrices.

We want to pick a permutation of cyclic type μ. We're going to take a *particular* permutation of this cyclic type — a ν_1 cycle acting on $1 \to 2 \to \cdots \to \nu_1 \to 1$, and then a cycle $\nu_1 + 1 \to \cdots \to \nu_1 + \nu_2 \to \nu_1 + 1$, and so on. We can easily express this permutation in terms of generators — the first cycle is $(s_1 s_2 \cdots s_{\nu_1 - 1})$, then we skip s_{ν_1} and get $s_{\nu_1+1} s_{\nu_1+2} \cdots s_{\nu_2 - 1}$, and so on.

Then we can take the product of the corresponding matrices R_{s_i}, and our goal is to calculate the trace of the matrix. We know $R_w \colon v_T \to \cdots$; to find $\operatorname{tr} R(w)$, we want to sum over all T the coefficient of v_T in $R_w(v_T)$.

We claim that when we calculate this trace, we can forget about the $v_{\widetilde{T}}$ term. This is only really true for our particular choice of w. The idea is that we first act by s_1, then s_2, then so on — so we start with some SYT, and then either keep it or swap 1 and 2, then keep it and swap 2 and 3, or so on. In the end we want the original tableaux; to have that, we cannot ever perform a switch (because if we perform a switch, we will never be able to get back to the original tableaux).

So then our result will be a sum over SYT of the products $1/(c_{i+1} - c_i)$, and we can explicitly simplify this expression to get our formula.

§27 November 16, 2022

§27.1 Murnaghan–Nakayama Rule

Recall that λ/μ is a skew shape with n boxes. For this skew shape, we can define a representation $V_{\lambda/\mu}$ of S_n (which is not necessarily an irreducible representation) given by the same formulas as Young's orthogonal form. In other words, S_n acts on the linear space over $\mathbb{C}$ with basis $\{v_T\}$ where $T \in \mathrm{SYT}(\lambda/\mu)$, where each generator s_i acts by the matrix

$$R_{s_i} \colon v_T \mapsto \frac{1}{c_{i+1} - c_i} \cdot v_T + \sqrt{1 - \frac{1}{(c_{i+1} - c_i)^2}} \cdot v_{\tilde{T}}.$$

(The second term only applies in the second case.) Here c_i is the *content* of the box with entry i in T (in other words, the diagonal), and $\tilde{T}$ is obtained by switching i and $i+1$.

This is a matrix whose rows and columns are labelled by these standard Young tableaux; it's not hard to see that these matrices satisfy the Coxeter relations.

Now suppose $\nu = (\nu_1, \ldots, \nu_k)$ is a composition of n (it can be a partition, but we don't require that the parts are arranged in weakly decreasing order). Then we define $\chi_{\lambda/\mu}(\nu)$ as the value of the character of $V_{\lambda/\mu}$ on a permutation $w \in S_n$ of cyclic type ν.

Theorem 27.1 (Murnaghan–Nakayama Rule)
We have

$$\chi_{\lambda/\mu}(\nu) = \sum_{\mathrm{RT}} (-1)^{\mathrm{ht}(T)}$$

where the sum is over all ribbon tableaux RT of shape λ/μ and type ν.

Recall that a ribbon tableaux is like a standard Young tableaux, but made out of ribbons instead: for example,

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

In particular every ribbon is connected — for example

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

is not a ribbon tableaux.

Recall that to find the height, the height of a *ribbon* is $a - 1$ where the ribbon has a rows, and the height of the ribbon tableaux is the sum of the heights of its ribbons.

Somehow, this rule looks asymmetric with respect to the conjugation of shapes — if we reflect the shape with respect to the diagonal, it looks like we get a different rule. But in fact, it is symmetric:

Lemma 27.2

We have $V_{(\lambda/\mu)'} = V_{\lambda/\mu} \otimes \mathrm{sgn}$, where sgn is the sign representation of S_n (corresponding to $(1, 1, \dots, 1)$). In particular,

$$\chi_{(\lambda/\mu)'}(w) = \chi_{\lambda/\mu}(w) \cdot \mathrm{sgn}(w).$$

This follows easily from Young's orthogonal form. But if we conjugate the shape, then the heights of ribbons become *widths* of ribbons ($b-1$, where the ribbon has b columns). We also have the extra factor of the sign; the sign of a n-cycle is $(-1)^{n-1}$, so the sign of any permutation of cyclic type ν is $(-1)^{\sum(\nu_i - 1)}$.

The main point is that there's a relation between a, b, and the size of the ribbon: we can see that

$$(a-1) + (b-1) = (r-1),$$

where the ribbon has r boxes. So then multiplying by the sign of the permutation transforms heights into widths, and that means this rule is basically symmetric with respect to switching rows and columns.

Corollary 27.3

If λ/μ is a self-conjugate shape (so that $(\lambda/\mu)' = \lambda/\mu$), then for any permutation w with $\mathrm{sgn}(w) = -1$, we have $\chi_{\lambda/\mu}(w) = 0$.

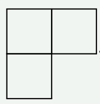
This is because we know

$$\chi_{(\lambda/\mu)'}(w) = \chi_{\lambda/\mu}(w) \cdot \mathrm{sgn}(w) = \chi_{\lambda/\mu}(w) \cdot (-1),$$

so we must have character 0.

Example 27.4

Last class, we calculated that for the Young diagram

$\lambda =$  ,

we got $\chi(\mathrm{Id}) = 2$, $\chi(s_1) = 0$, and $\chi(s_1 s_2) = -1$.

§27.2 Proof of the Murnaghan–Nakayama Rule

We'll actually prove an even more general version of the rule. But first, let's try to unfold this definition and get a combinatorial formula for the characters.

Suppose that $\nu = (\nu_1, \dots, \nu_k)$. We're allowed to take any permutation of cyclic type ν; we then want to calculate the trace of R_w by writing w in terms of the generators. The simplest thing we can do is to take w to be a product of cycles — we have one cycle $(1, 2, 3, \dots, \nu_1)$, another cycle $(\nu_1 + 1, \dots, \nu_2)$, and so on. This is easy to write down in terms of the simple generators — we have

$$w = (s_1 s_2 \cdots s_{\nu_1 - 1})(s_{\nu_1+1} s_{\nu_1+2} \cdots s_{\nu_1+\nu_2-1}) \cdots.$$

So we essentially have the consecutive generators, but with some entries skipped — we write all the adjacent transpositions in order, and we skip s_{ν_1}, $s_{\nu_1+\nu_2}$, $\dots$, $s_{\nu_1+\cdots+\nu_k} = s_n$ (we always skip s_n because s_n does not exist).

We will actually take the *inverse* permutation (writing down this product in the opposite order) for convenience. So then we have

$$\chi_{\lambda/\mu}(\nu) = \chi_{\lambda/\mu}(w^{-1}) = \operatorname{tr}(\cdots \hat{R}_{s_{\nu_1+\nu_2}} \cdots \hat{R}_{s_{\nu_1}} R_{s_2} R_{s_1}).$$

By definition, the trace is

$$\sum_{T \in \mathrm{SYT}(\lambda/\mu)} \text{coefficient of } v_T \text{ in } (\cdots R_{s_1})(v_T)$$

(since such T index our matrices).

We can see that for every factor, there are two options — we take this tableaux and either go to itself, or go to the tableau obtained by switching i and $i+1$. So at the first stage we leave it or switch 1 and 2, at the second we leave it or switch 2 and 3, and so on. In the end we want to get the original tableau. But if at any stage we choose the swapped term, we can never get back to the original — this means we can ignore the weird square root terms, and we'll only get the relatively simple expressions $1/(c_{i+1} - c_i)$. This is the crucial observation:

Claim 27.5 (Observation) — Only the diagonal terms

$$\frac{1}{c_{i+1} - c_i} v_T$$

make a contribution to the trace.

So now we can combine everything together, to get the following expression for $\chi_{\lambda/\mu}(w^{-1})$:

Proposition 27.6
We have

$$\chi_{\lambda/\mu}(\nu) = \sum_{T \in \mathrm{SYT}(\lambda/\mu)} \prod_{i \in [n] \setminus \{\nu_1, \nu_1+\nu_2, \ldots\}} \frac{1}{c_{i+1} - c_i},$$

where c_i (depending on T) is the content of box i in T.

So we take some SYT and consider the box i, and its diagonal is c_i.

This is what we get from essentially just unfolding the definition of the characters. Somehow, we want to see that this sum over SYT magically converts into a sum of ribbon tableaux.

Now we'll write down the Murnaghan–Nakayama rule in the special case where our permutation is a n-cycle — meaning that $\nu = (n)$.

Theorem 27.7 (Special Case of M–N Rule)
We have

$$\sum_{T \in \mathrm{SYT}(\lambda/\mu)} \frac{1}{c_2 - c_1} \cdot \frac{1}{c_3 - c_2} \cdots \frac{1}{c_n - c_{n-1}} = \begin{cases} (-1)^{\mathrm{ht}(\lambda/\mu)} & \text{if } \lambda/\mu \text{ is a ribbon} \\ 0 & \text{otherwise.} \end{cases}$$

It's actually easy to see that this special case of the M–N rule implies the general case:

Claim — The identity (*) implies the general case of the M–N rule.

Proof. We've already seen that

$$\chi_{\lambda/\mu}(\nu) = \sum_{T \in \mathrm{SYT}(\lambda/\mu)} \prod_{i \in [n] \setminus \{\nu_1, \nu_1+\nu_2, \ldots\}} \frac{1}{c_{i+1} - c_i}.$$

This expression can be rewritten: suppose we have some T of shape λ/μ. Then we can first look at the part of the tableaux filled with 1 to ν_1, and then $\nu_1 + 1$ to $\nu_1 + \nu_2$, and so on — so we decompose our tableaux into k smaller standard Young tableaux $T_1, T_2, \dots, T_k$ of skew shapes. Then we get

$$\chi_{\lambda/\mu}(\nu) = \sum_{\mu \subseteq \lambda_1 \subseteq \cdots} \left(\sum_{T_1 \in \mathrm{SYT}(\lambda^{(1)}/\mu)} \frac{1}{c_2 - c_1} \cdots \right) \left(\sum_{T_2 \in \mathrm{SYT}(\lambda^{(2)}/\lambda^{(1)})} \right)$$

where the sum is over all sequences $\mu \subseteq \lambda_1 \subseteq \lambda_2 \subseteq \cdots \subseteq \lambda_k = \lambda$, where each box has ν_i more boxes than the previous. These expressions are all 0 unless the T_i are ribbons, so then the outer sum reduces to a sum over ribbon tableaux, and applying the special case gives our desired result. $\square$

Example 27.8

For the shape $\begin{array}{|c|c|}\hline & \\ \hline \end{array}$, we have only one SYT

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

which has

$$\frac{1}{c_2 - c_1} = 1.$$

Meanwhile, for the vertical domino we have one SYT

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

which has

$$\frac{1}{c_2 - c_1} = -1.$$

We can see that the height of the first is 0, and the height of the second is 1.

Example 27.9

For the shape

$$\lambda/\mu = \begin{array}{cc} \cline{2-2} & \multicolumn{1}{|c|}{} \\ \hline \multicolumn{1}{|c|}{} & \\ \cline{1-1} \end{array},$$

this shape is not a ribbon, so (*) says that we should get 0. There are two tableaux

$$\begin{array}{cc} \cline{2-2} & \multicolumn{1}{|c|}{2} \\ \hline \multicolumn{1}{|c|}{1} & \\ \cline{1-1} \end{array} \text{ and } \begin{array}{cc} \cline{2-2} & \multicolumn{1}{|c|}{1} \\ \hline \multicolumn{1}{|c|}{2} & \\ \cline{1-1} \end{array},$$

which gives us

$$\frac{1}{2} + \frac{1}{-2} = 0.$$

Whenever the shape is disconnected, we should get 0.

Example 27.10

The shape $\lambda = \begin{array}{|c|c|} \hline & \\ \hline & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}$ has two SYT

$$\begin{array}{|c|c|} \hline 1 & 2 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \text{ and } \begin{array}{|c|c|} \hline 1 & 3 \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}.$$

Each expression involves two factors

$$\frac{1}{(c_2 - c_1)(c_3 - c_2)}.$$

We then end up with

$$\frac{1}{1(-2)} + \frac{1}{(-1)(2)} = -1,$$

which agrees with the formula because this ribbon has height 1.

Example 27.11

Suppose our shape is a 2×2 box, with two SYT

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

Then we get products of three factors — we end up with

$$\frac{1}{1} \cdot \frac{1}{-2} \cdot \frac{1}{1} + \frac{1}{-1} \cdot \frac{1}{2} \cdot \frac{1}{-1} = 0.$$

This is not a ribbon — so whenever our shape is disconnected or contains a 2×2 box, we should get 0, and when it's a ribbon we should get ± 1.

But it's actually easier to generalize this identity — we can actually replace the contents by arbitrary variables (which makes it easier to see why the identity holds).

Definition 27.12. Fix a collection of variables $x = (x_i)_{i \in \mathbb{Z}}$. We define a multivariate generalization of character values by

$$\chi^x_{\lambda/\mu}(\nu) = \sum_{\mathrm{SYT}(\lambda/\mu)} \prod_{i \in [n] \setminus \{\nu_1, \nu_1 + \nu_2, \ldots\}} \frac{1}{x_{c_i(T)} - x_{c_{i+1}(T)}}.$$

We get back to the original case if $x_c = -c$ for all $c \in \mathbb{Z}$.

Theorem 27.13 (Multivariate Generalization of MN)

We have

$$\chi^x_{\lambda/\mu}(\nu) = \sum_{\mathrm{RT}} \mathrm{wt}(T)$$

(where the sum is over all ribbon tableaux of shape λ/μ and type ν).

To define the weight of a tableaux, we'll first define the weight of a ribbon: we know the contents of the boxes in a ribbon form a consecutive set c, $c+1$, $\ldots$, c'. Then we define

$$\mathrm{wt}(\mathrm{ribbon}) = (-1)^{\mathrm{ht}} \prod_{i \in [c, c'-1]} \frac{1}{x_i - x_{i+1}},$$

and we define wt(RT) as the product of weights of its ribbon.

For the same reason as above, it's enough to show this for the case where ν has a single cycle, so we will concentrate on this case.

Example 27.14

Suppose that

$$\lambda/\mu = \begin{ytableau} \none & {} & {} \\ {} & {} \end{ytableau},$$

so the standard Young tableaux are

$$\begin{ytableau} \none & 2 & 4 \\ 1 & 3 \end{ytableau}, \quad \begin{ytableau} \none & 2 & 3 \\ 1 & 4 \end{ytableau}, \dots.$$

For the sake of indexing, we'll label the contents by 1, 2, 3, 4 instead of -1, 0, 1, 2. Then we get

$$\frac{1}{x_1 - x_3} \cdot \frac{1}{x_3 - x_2} \cdot \frac{1}{x_2 - x_4} + \frac{1}{x_1 - x_2} \cdot \frac{1}{x_2 - x_3} \cdot \frac{1}{x_1 - x_3} \cdot \frac{1}{x_3 - x_4} \cdot \frac{1}{x_4 - x_2} + \cdots.$$

All of these expressions have a simple shape — we have a sort of chain of indices — and if we add them together, then we end up with

$$-\frac{1}{x_1 - x_2} \cdot \frac{1}{x_2 - x_3} \cdot \frac{1}{x_3 - x_4}.$$

§28 November 18, 2022

§28.1 Murnaghan–Nakayama Rule

Recall that we fixed (x_i) for each $i \in \mathbb{Z}$, and we defined the generalized character

$$\chi^x_{\lambda/\mu}(x) := \sum_{T \in \mathrm{SYT}(\lambda/\mu)} \prod_{i \in [n] \setminus \{\nu_1, \nu_1 + \nu_2, \dots\}} \frac{1}{x_{c(i)} - x_{c(i+1)}}.$$

Theorem 28.1

We have

$$\chi^x_{\lambda/\mu}(\nu) = \sum_{\mathrm{RT}} \mathrm{wt}(T),$$

where the sum is over all ribbon tableaux of shape λ/μ and type ν, the weight of a ribbon tableaux is the product of the weight of its ribbons, and the weight of a ribbon spanning contents c to c' is

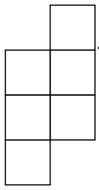
$$(-1)^{\mathrm{ht}(T)} \prod_{i \in [c, c'-1]} \frac{1}{x_i - x_{i+1}}.$$

Then taking $x_i = -i$ for all i gives us the usual characters. (This is what we get essentially from expanding the definition of Young's orthogonal form — in Young's orthogonal form we had coefficients of $\frac{1}{a-b}$, which is where this comes from.)

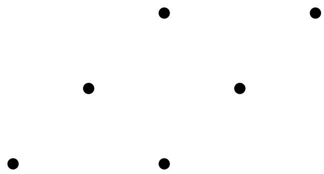
It helps to understand the MN rule in this setting. First, what's special about ribbons? To see that, it helps to generalize this theorem to an even more general class of shapes.

§28.2 Abacus Drawings

Suppose we have a skew shape

$\lambda/\mu =$  .

Then in its *abacus drawing*, we rotate by 45° so that the diagonals become vertical strings, and the boxes become little balls on these strings:



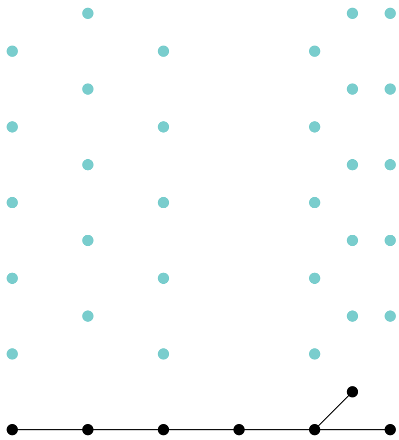
We'll generalize this to an even broader type of shape. We'll consider a tree $\mathbb{T}$ (possibly infinite) — in the case of skew shapes, $\mathbb{T} = \mathbb{Z}$ is the infinite chain in both directions.

To this tree, we associate an **abacus poset** $\mathcal{P}_{\mathbb{T}}$ as follows:

Example 28.2

Consider the tree

Above every node, we draw a string of nodes, such that for any two neighboring nodes the strings are interlaced.



For a node $a \in \mathcal{P}_{\mathbb{T}}$, we use $c(a)$ to denote the corresponding node in $\mathbb{T}$ (which we'll think of as the content).

Definition 28.3. A **shape** κ is a finite subset of $\mathcal{P}_{\mathbb{T}}$ with n elements which is convex — if a and b belong to κ, then all elements $x \in [a, b]$ are also in κ.

If our tree is just $\mathbb{Z}$, then such convex shapes correspond to skew Young diagrams. In general, you can imagine something like skew Young diagrams, but they can branch into several layers (where the tree has a branching).

Definition 28.4. $\mathrm{SYT}(\kappa)$ is the set of linear extensiosn of κ.

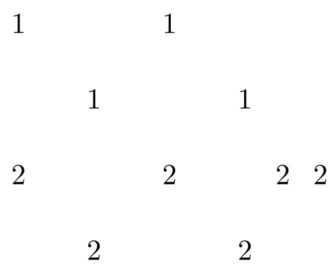
A linear extension is a way of labelling the nodes our shape from 1 to n such that the labelling agrees with the ordering.

Definition 28.5. A **ribbon** is a shape κ such that the Hasse diagram of κ is a directed tree.

Definition 28.6. A **tree tableau** of shape κ and type ν is a labelling $\mathrm{TT}\colon \kappa \to \{1, \ldots, k\}$ such that:

- The labels weakly increase (i.e., they agree with the poset);
- For all $i \in [k]$, the set $\{a \in \kappa \mid \mathrm{TT}(a) = i\}$ is a directed tree with ν_i elements.

For example, for the following orange shape:



Definition 28.7. The weight of a tree tableaux is the product of weights of its trees, and the weight of a tree is

$$\prod_{i \to j \in T} \frac{1}{x_{c(i)} - x_{c(j)}}.$$

We assume our edges are always directed downwards. So in the above tree tableau, the 1-tree would give us

$$\frac{1}{x_1 - x_2} \cdot \frac{1}{x_3 - x_2} \cdot \frac{1}{x_3 - x_4}.$$

We'd have a similar factor corresponding to the second tree, of

$$\frac{1}{x_1 - x_2} \cdot \frac{1}{x_3 - x_2} \cdots.$$

(For every edge in the tree, we look at the corresponding variables, and take their difference.) In the classical case, this would specialize to exactly what we need.

Theorem 28.8

We have the identity

$$\chi_\kappa^x(\nu) = \sum_{\mathrm{TT}} \mathrm{wt}(\mathrm{TT})$$

over all tree tableaux of shape κ and type ν, where χ_κ^x is defined as before (as a sum over all SYT of shape κ).

As before, it's enough to prove this in the case where ν consists of a single part.

Theorem 28.9

We have

$$\chi_\kappa((n)) = \begin{cases} \mathrm{wt}(\kappa) & \text{if } \kappa \text{ is a tree} \\ 0 & \text{otherwise.} \end{cases}$$

This easily follows from three lemmas. First, it'll be convenient to use the following notation: our variables are labelled by nodes of $\mathbb{T}$, but for convenience we'll use integers.

Notation 28.10. For any sequence $i_1, \ldots, i_\ell$, we use

$$\langle i_1, \ldots, i_\ell \rangle = \frac{1}{x_{i_1} - x_{i_2}} \cdots \frac{1}{x_{i_{\ell-1}} - x_{i_\ell}}.$$

So our theorem is an identity involving such expressions. It follows from three identities:

Lemma 28.11 (Lemma 1)

If A and B are nonempty disjoint subsets of $\mathbb{Z}$, then

$$\sum_{c \in \mathrm{Shuffle}(A,B)} \langle c \rangle = 0.$$

In a shuffle, we take the elements of A and B, and preserve their internal order but can mix A and B.

Example 28.12

If $A = \{1\}$ and $B = \{2, 3\}$, then there are three possible shuffles of A and B, and this tells us

$$\langle 1, 2, 3 \rangle + \langle 2, 1, 3 \rangle + \langle 2, 3, 1 \rangle = 0.$$

(This will be an exercise on the next problem set.)

Our expression χ_κ^x is a sum over linear extensions of such things. If κ is disconnected, then the labellings of the two parts are sort of independent. So this expression becomes a sum over shufflings of the labels, which gets this case.

Lemma 28.13 (Lemma 2)

Suppose $\{1\}$, A, and B are disjoint and nonempty. Then

$$\sum_{c \in \mathrm{Shuffle}(A,B)} \langle 1, c, 1 \rangle = 0.$$

Example 28.14

When $A = \{2\}$ and $B = \{3, 4\}$, this tells us that

$$\langle 1, 2, 3, 4, 1 \rangle + \langle 1, 3, 2, 4, 1 \rangle + \langle 1, 3, 4, 2, 1 \rangle = 0.$$

This explains the situation where the Hasse diagram of κ contains a cycle. One can show that whenever we have a cycle, we'll have two elements with the same content (for this particular type of poset). We'll then have intermediate nodes between them appearing in different branches, which we can shuffle; that means this lemma tells us we get 0 whenever we have a cycle.

In other words, one can show that if we have a cycle, then there's two elements on the same string:

1

3 5

4

5

This linear extension would give us $\langle 2, 4, 1, 3, 2\rangle$. But we can see that between 1 and 5, some labels are on the left side, and the others are on the right side; and we can shuffle these labels as long as we preserve their order. So we'll get a sum over all shuffles of these labels, which will be automatically 0.

So it remains to show that when κ is a tree we get what we want.

Lemma 28.15 (Lemma 3)
If T is a directed tree, then

$$\sum_{w_1\cdots w_n} \langle w_1, \ldots, w_n\rangle = \prod_{i\to j \text{ edge of } T} \frac{1}{x_i - x_j}.$$

where the sum is over all linear extensions of T.

Example 28.16
For a tree $a \to b \to c, d$, we have two linear extensions — $\langle a, b, c, d\rangle$ and $\langle a, b, d, c\rangle$. This should equal

$$\frac{1}{x_a - x_b}\frac{1}{x_b - x_c}\frac{1}{x_b - x_d}.$$

These are certain identities for rational expressions, and all of them follow from one key identity.

The key identity is the following:

$$\frac{1}{x_i - x_j} \cdot \frac{1}{x_j - x_k} = \frac{1}{x_i - x_k} \cdot \frac{1}{x_i - x_j} + \frac{1}{x_i - x_k} \cdot \frac{1}{x_j - x_k}.$$

We can graphically represent this by drawing an edge $i \to j$ to represent such a term, and repeatedly applying it will give all three lemmas.

This gives rise to the *Orlik–Terao algebra* (for type A braid arrangements).

Definition 28.17. The Orlik–Terao algebra OT_n is the commutative algebra with generators a_{ij} for $i, j \in [n]$, such that $a_{ij} = -a_{ji}$ and

$$a_{ij} \cdot a_{jk} = a_{ij}a_{ik} + a_{ik}a_{jk}.$$

These a_{ij} are secretly equal to $1/(x_i - x_j)$, so the algebra is isomorphic to the algebra generated by such terms.

All the relations we've written so far can be written as relations in the Orlik–Terao algebra. There are also other amazing identities:

Suppose we start witih a chain $1 \to 2 \to 3 \to 4$, where every edge corresponds to a_{ij} — so this graph represents $a_{12}a_{23}a_{34}$. Whenever se see two edges forming a 2-chain, we can replace them by either the first or second expression — so our graph gives us two graps $1 \to 2$, $1 \to 3 \to 4$ and $1 \to 3 \to 4$, $2 \to 3$. Then we can take another pair and replace them again — for example, in the first we can take $1 \to 3 \to 4$ and replace it with $1 \to 4$, $1 \to 3$ or $1 \to 4$, $3 \to 4$. We keep oin going until we are stuck.

In our example, we have 5 endpoints.

Claim — If we start with a n-chain, the number of graphs we end up with is the $(n-1)$th Catalan number.

Note that there's many ways to play the game — we could pick any two-chain and apply this rule, and we might end up with a different set of graphs in the end, but we'll end up with the same *number* of them.

This is a certain identity in the Orlik–Terao algebra. Another is the following:

Claim — If we start with K_n, then the number of endpoints we end up with is $C_{n-1}C_{n-2}\cdots C_1$.

For an n-chain, this is relatively easy to see. But for the complete graph, there is no known combinatorial proof.

§29 November 21, 2022

§29.1 Orlik–Solomon and Orlik–Terao Algebras

At the end of last class, we mentioned the Orlik–Terao algebra. There's another even more well-known algebra, the Orlik–Solomon algebra. We'll talk about these algebras for *type A braid arrangements*.

Remark 29.1. The Orlik–Solomon algebra should be referred to as the Arnold–Orlik–Solomon algebra, as Arnold defined them for this case first.

These two algebras are very similar, but the main difference is that OS_n is *anti-commutative*, while OT_n is commutative.

OS_n is generated (over $\mathbb{C}$) by the generators a_{ij} for $i, j \in [n]$, where $a_{ij} = a_{ji}$. Meanwhile, OT_n is generated (over $\mathbb{C}$) by c_{ij} for $i, j \in [n]$, with $c_{ij} = -c_{ji}$. So we can always assume $i \le j$ in the indices, but it's often convenient to have the extensions to all index pairs.

In OS_n we have $a_{ij}a_{k\ell} = -a_{k\ell}a_{ij}$ (anti-commutativity), and $a_{ij}^2 = 0$. Similarly in OT_n we have $c_{ij}c_{k\ell} = c_{k\ell}c_{ij}$.

The main relation in both cases is the same:

$$a_{ij}a_{jk} = a_{ik}a_{ij} + a_{jk}a_{ik} \text{ and } c_{ij}c_{jk} = c_{ik}c_{ij} + c_{jk}c_{ik}.$$

We can identify the generators with edges:

Theorem 29.2 (Arnold)

The cohomology $H^*(\mathbb{C}^n \setminus \{x_i - x_j \mid i < j\}, \mathbb{C})$ is isomorphic to OS_n.

Example 29.3

In the case $n = 3$, we take $\mathbb{C}^3$ and remove three hyperplanes. The picture is 3-dimensional, but we can always take the 2-dimensional section where the sum of all coordinates is zero — so then we get the two-dimensional complex space. We then take the complement of these hyperplane arrangements. In the real case, this would break into regions; but in the complex plane, each of these lines would have codimension 2. Then you can look at the topology, and these things generate the cohomology.

There is a model for cohomology in terms of differential forms. In this model,

$$a_{ij} \rightsquigarrow \frac{d(x_i - x_j)}{x_i - x_j}$$

for the Orlik–Solomon algebra, and for the Orlik–Terao algebra

$$c_{ij} \rightsquigarrow \frac{1}{x_i - x_j}.$$

§29.1.1 Hilbert Series

Question 29.4. What are the Hilbert series $\sum_{k\ge 0} \dim \mathrm{OS}_n^k \cdot t^k$, where OS_n^k is the kth graded component of OS_n?

We can ask the same question for OT_n. In fact, the series for OS_n is a polynomial, since it's finite-dimensional — anything which is not squarefree becomes 0.

Example 29.5

When $n = 3$, we have:

k	0	1	2	3
$\dim \mathrm{OS}_n^k$	1	3	2	0
$\dim \mathrm{OT}_n^k$	1	3	5	

When $k = 0$ we just have 1. When $k = 1$ we have three linearly independent generators, so the dimension is 3. When $k = 2$ we want to look at monomials of degree 2 in the generators — we have $a_{12}a_{23}$, $a_{12}a_{13}$, and $a_{23}a_{12}$, but they are not linearly independent — we have one relation between them, so we have a 2-dimensional space.

In OT_n, in this case we have more because we can take the squares — this gives us dimension 5.

When $k = 3$, the only monomial with three generators is $a_{12}a_{13}a_{23}$. But this is actually 0, since applying the relation gives us something with squares. But in OT_n we get nonzero terms.

In this example, we have

$$\mathrm{Hilb}_{\mathrm{OS}_3}(t) = 1 + 3t + 2t^2 = (1+t)(1+2t).$$

From this example you can guess the general answer:

Theorem 29.6

We have

$$\mathrm{Hilb}_{\mathrm{OS}_n}(t) = (1+t)(1+2t)\cdots(1+(n-1)t).$$

SUppose that we've already found a linear basis for OS_n^k. From this we can also construct a linear basis for OT_n — the only difference is that we're not restricted to the square-free case. So if we can use a_{ij} in OS_n, then in the basis we can use any power of it. So here instead of t (representing one generator) we should have $t + t^2 + t^3 + \cdots$.

Theorem 29.7

We have

$$\mathrm{Hilb}_{\mathrm{OT}_n}(t) = \mathrm{Hilb}_{\mathrm{OT}_n}\left(\frac{t}{1-t}\right) = \frac{1}{(1-t)^{n-1}}(1+t)(1+2t)\cdots(1+(n-1)t).$$

Exercise 29.8. Prove this theorem (with both parts).

We'll see two possible approaches. To calculate the Hilbert series, we would like to present a collection of monomials that form a linear basis.

First of all, we want to find a linear basis of OS_n. We can notice that if we have any square we automatically get 0. If we have $a_{12}a_{13}a_{23}$, we also automatically get 0.

We can represent any monomial by a graph. From the relations, we can show that whenever we have a cycle, we get 0. So our basis will consist of some graphs without cycles — meaning forests.

But not all forests are linearly independent, so we have to present some set of forests that are linearly independent. For example, our three trees in the above relation are linearly independent. We can modify the relation slightly to get the following game on graphs: whenever we have a graph with a pair of edges of the form $i \to k$ and $j \to k$ with $i < j < k$ (we assume edges are always directed from the smaller to larger vertex), we can remove these two edges and replace them with $i \to j \to k$ and $i \to k$ and $i \to j$. (We should also care about the order in which we write down the edges — but we don't actually care about the signs, since we can just try to construct a basis up to signs.)

Now we can start with a graph, and whenever we see a pair of edges like this, we play the game. In the end, every monomial will reduce to some alternating sum of monomials in the a_{ij} corresponding to graphs without a pair of edges $i \to k$ and $j \to k$.

Note that this is a bit different than the game from last lecture.

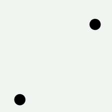
Claim — No matter how we play this game, we'll get the same result, and the resulting graphs form a basis of OS_n.

These resulting graphs have a name:

Definition 29.9. $F \subseteq K_n$ is a **increasing forest** if each connected component has a *root* which equals its minimal element, and all labels are increasing as we go away from the root.

Example 29.10

The graph



We can check that the set of graphs without the forbidden pattern is exactly the set of increasing trees:

Claim — The monomials corresponding to increasing forests in K_n form a linear basis of OS_n.

For OT_n, we can get a closely related result by looking at the same set of forests, but allowing ourselves to have any number of copies of any edge.

Remark 29.11. This is not the only basis of OS_n. Another choice is to require that in our forest, each connected component is a chain with root starting at the minimal element. For example, if the graph has only one connected component, then it would have to be a chain $1 \to 5 \to 3 \to 4 \to 6$, for instance.

Now the calculation of Hilbert series reduces to a calculation fo the number of such forests. So in particular we get

$$\sum_{F \subseteq K_n} t^{\#\text{edges in } F} = (1+t)(1+2t)\cdots(1+(n-1)t) = \sum_{k=0}^{n-1} c(n, n-k)t^k.$$

There is a special name for these things: they are called **signless Stirling numbers** of the first kind, and by definition they are equal to the number of permutations in S_n with exactly $n-k$ cycles. So essentially, increasing forests counted by the number of connected components counts the number of such permutations; the same is true for the other basis.

We have a similar basis for OT_n, except that you can take graphs with multiple edges.

§29.2 Another Game on Graphs

Recall that in the previous lecture, we played *another* game on graphs. We are now going to look at OT_n and play a similar but different game.

Graphs will represent monomials in OT_n. (In particular, everything is commutative.)

In this game, if we see a graph with a pair of edges $i \to j \to k$ with $i < j < k$, we are allowed to replace this graph by two graphs — G', which has $i \to k$ and $i \to j$, and G'', which has $i \to k$ and $j \to k$.

We start with any graph, and we do this operation; if we find another 2-chain, then we repeat, and so on. We do this until we get a linear combination of some graphs without any 2-chains, called **alternating graphs** (a directed graph where all edges are directed from the smaller to larger vertex, and there is no 2-chain — this means in any path in this graph, the labels alternate, and equivalently every vertex is a source or a sink).

Unlike in the previous game (where the result is the same no matter how we play the game), the result will be different for different ways to play the game. But the number of graphs we get will be the same, and this is a fairly nontrivial number.

Question 29.12. What if our original graph is a tree?

Then when we apply this transformation, all the graphs we get as a result will still be trees.

Assume that G is a tree.

Example 29.13

Suppose G is a chain $1 \to 2 \to \cdots \to n$. Then one way to play the game will give us all alternating non-crossing trees in K_n — *alternating* means there are no 2-chains, and *non-crossing* means we don't have edges that cross if we arrange all vertices on a line (i.e. $i < j < k < \ell$ with edges $i \to k$ and $j \to \ell$).

Meanwhile, there is also another way to play this game, which will give us all alternating non-*nesting* trees (where non-nesting means that we don't have $i \to \ell$ and $j \to k$).

Example 29.14

When $n = 4$, the alternating non-crossing trees are $1 \to 2$, $1 \to 3$, $1 \to 4$; $1 \to 2$, $1 \to 4$, $3 \to 4$; $1 \to 4$, $2 \to 4$, $3 \to 4$; $1 \to 4$, $1 \to 3$, $2 \to 3$; and $1 \to 4$, $2 \to 4$, $2 \to 3$. This is one way to play the game.

Another way to play teh game will give us the first three, as well as $1 \to 3$, $2 \to 3$, $2 \to 4$ and $1 \to 4$, $1 \to 3$, $2 \to 4$.

There are a bunch of interpretations of Catalan numbers, some of which involve non-crossing things and some of which involve non-nesting things; this is an example of that phenomenon.

§29.3 A Geometric Interpretation

This game has a nice geometric interpretation in terms of polytopes — for a graph G, we can construct a **root polytope** R_G. We'll do this for trees.

This polytope is defined as the convex hull of the origin and all $e_i - e_j$ for all edges $i \to j$ in the graph (with $i < j$) — so we have a convex polytope assigned to every tree.

Claim — Whenever we play this game, the polytope for our original graph decomposes into a union of the polytopes of the two resulting graphs, which have non-overlapping interiors.

So we essentially take a polytope R_G, and then it breaks into two pieces $R_{G'}$ and $R_{G''}$ in any way we play the game. We keep subdividing, and the end result will correspond to a triangulation of the polytope. So this is one choice of a triangulation, and the other is another. The point is that any final piece will be a unit simplex — a simplex witih the same volume. So the theorem that the number of terms doesn't depend on the way you play the game means that any triangulation has the same number of simplices, although the exact collection of simplices may be different.

§30 November 23, 2022

§30.1 Root Polytopes

Last class, we saw a game on graphs, where given a graph G with a two-chain $i \to j \to k$, we can replace it with a graph G' with $i \to k$ and $i \to j$, and another graph G'' with $i \to k$ and $j \to k$.

In some cases, this operation can be viewed in terms of polytopes — the operation of replacing a graph with these two graphs can be thought of as breaking a polytope into two smaller polytopes.

Definition 30.1. Suppose G is a directed graph on vertex set $[n]$. The **root polytope** of G is the convex hull

$$R_G := \operatorname{Conv}(0, e_i - e_j \text{ for}(i, j) \in E).$$

These vectors are called *roots* (of a type A root system).

There are three special cases of root polytopes which are interesting.

Example 30.2

Consider the case when G is the complete *bidirected* graph (the complete graph where for every $i \neq j$ we have an edge $i \to j$ and an edge $j \to i$).

Example 30.3

When $n = 3$, the root polyope is a hexagon.

The second special case is when we only have an edge $i \to j$ for $i < j$ — so instead of taking all roots, we're only taking positive roots.

Example 30.4

When $n = 3$, our root polytope consists of the origin and three points (so it is a rhombus).

A third special case is when G is a complete bipartite graph $K_{k,n-k}$, where all edges are directed from one part to the other — so the edges are (i, j) for $i \in \{1, \ldots, k\}$ and $j \in \{k + 1, \ldots, n\}$.

Example 30.5

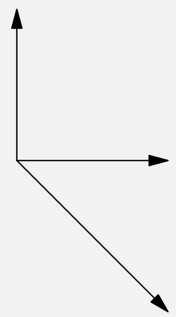
The root polytope for $K_{1,2}$ is a little triangle (with the bottom vertex at the origin, and two vertices at the top).

Question 30.6. What is the volume fo the root polytope?

An even more basic question is how to define the volume — all these polytopes are in $\mathbb{R}^n$, but they're actually $(n-1)$-dimensional (they belong to the hyperplane with sum 0). But no matter how we define the volume, some comparisons are obvious — the volume of the second example is twice that of the third, and the volume of the first is six times.

The easiest way to define the volume is the following: consider the projection map $p\colon \mathbb{R}^n \to \mathbb{R}^{n-1}$ sending $(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_{n-1})$. Then $\operatorname{Vol} R_G$ is defined as the volume of its projection — as $\operatorname{Vol}_{n-1} p(R_G)$.

Remark 30.7. Our pictures are in 3D space, but we're drawing them on the plane $x + y + z = 0$. If we projected them, we'd get a different picture: $e_1 - e_2$ becomes $(1, -1)$ when we ignore the last coordinate, $e_1 - e_3$ becomes $(1, 0)$, and $e_2 - e_3$ becomes $(0, 1)$.



Then we can extend these lines in the reverse direction to get the full thing.

As we can see, this looks a bit deformed. It's not hard to show that if we ignore any other coordinate, we get the same volume; the picture will be deformed in a different way, but the volume would be the same.

Theorem 30.8

We have

$$(n-1)!\operatorname{Vol} R_{K_n^{\leftrightarrow}}\binom{2n-2}{n-1}$$

(for the complete bidirected graph),

$$(n-1)!\operatorname{Vol} R_{K_n^{\rightarrow}} = C_{n-1}$$

(for the complete unidirected graph), and

$$(n-1)!\operatorname{Vol} R_{K_{k,n-k}} = \binom{n-1}{k-1}.$$

So in these three special cases, we get some nice combinatorial expressions; this means it's natural to ask for a combinatorial interpretation of the volume.

We will assume that G is *transitively closed* — if we have an edge $i \to j$ and $j \to k$, then we also have an edge $i \to k$. For example, a graph with the edges $1 \to 2$ and $2 \to 3$ but not $1 \to 3$ would not be transitively closed; in this case, the root polytope would be a $120°$ isosceles triangle. We want to consider the case where the structure of the polytope is determined locally around the origin — in all our cases, we can see a nice triangulation where we can look at a small neighborhood of the origin and differentiate between the cases, but if we look at this polytope and the rhombus one, around the origin they look exactly the same, which means somehow we don't want to consider this case.

Notation 30.9. $\overline{G}$ denotes the transitive closure of G (whenever we have an edge $i \to j$ and $j \to k$, we add an edge $i \to k$).

The key lemma that explains the relationship between root polytopes and this game on graphs is the following:

Lemma 30.10

Let T be any directed tree on $[n]$, and suppose $i \to j$ and $j \to k$ are edges of T. Then we play the game where we replace this pair with teh pair $(i \to k, i \to j)$ and $(i \to j, j \to k)$, producing two trees T' and T'' (keeping all other edges the same). Then

$$R_{\overline{T}} = R_{\overline{T'}} \cup R_{\overline{T''}},$$

and the dimension of the intersection of $R_{\overline{T'}}$ and $R_{\overline{T''}}$ is strictly smaller than $n-1$.

Note that we're considering the root polytopes of the *closures* of our trees, not of the trees themselves.

In particular, this means

$$\operatorname{Vol} R_{\overline{T}} = \operatorname{Vol} R_{\overline{T'}} + \operatorname{Vol} R_{\overline{T''}}.$$

Example 30.11

Consider the tree $1 \to 2 \to 3$. Then the transitive closure will be $K_3^{\rightarrow}$, so $R_{\overline{T}}$ will be the rhombus. This breaks into two equilateral triangles; the top one is $R_{\overline{T'}}$ and the right one is $R_{\overline{T''}}$. They intersect, but their common face has a lower dimension.

Proof. We basically just need to unfold the definition of the convex hull —

$$\operatorname{Conv}(v_1, \dots, v_\ell) = \{a_1 v_1 + \cdots + a_\ell v_\ell \mid a_1, \dots, a_\ell \geq 0, a_1 + \cdots + a_\ell = 1\}$$

(by definition). Then we just have to apply this definition to the root polytopes. In particular,

$$R_{\overline{T}} = \{\cdots + a(e_i - e_j) + b(e_j - e_k) + \cdots \mid a, b \geq 0\}.$$

Then we change these two edges in two ways — we can write

$$a(e_i - e_j) + b(e_j - e_k) = \begin{cases} b(e_i - e_k) + (a-b)(e_i - e_j) & a \geq b \\ a(e_i - e_k) + (b-a)(e_j - e_k) & a \leq b \end{cases}$$

So these two cases give us two pieces of the polytope. □

So every time we play this game, the root polytope breaks into smaller pieces. We also want to see that in the end, all pieces have the same volume — this is easy, but an important property of the collection of vectors $e_i - e_j$, called *unimodularity*.

Lemma 30.12

The following are equivalent:

(1) T is a directed tree on $[n]$;

(2) R_T is a $(n-1)$-dimensional simplex;

(3) $(n-1)! \operatorname{Vol} R_T = 1$.

Note that here we are dealing with R_T and not the transitive closure. This property is called the *unimodularity* property — equivalently, the collection of vectors $e_i - e_j$ is unimodular.

We'll prove this by example.

Example 30.13

Take T to be the graph $1 \to 2$, $1 \to 3$, $1 \to 5$, $4 \to 5$. (The tree does not have to be alternating.)

This means we have $e_1 - e_2$, $e_1 - e_3$, $e_1 - e_5$, and $e_4 - e_5$. We claim that the graph is a tree if and only if this collection of vectors forms a basis — if we have a cycle then we have a linear relationship between the vectors. So T is a tree if and only if these vectors form a basis for our $(n-1)$-dimensional subspace.

But we also claim that in this situation, the volume of the corresponding polytope is 1, up to a sign. THis is because we can write

$$\pm(n-1)! \operatorname{Vol} R_T$$

as a determinant of a $(n-1) \times (n-1)$ matrix whose rows correspond to these vectors, where we write down the vectors in $\mathbb{R}^n$ and ignroe their last coordinate. So for example, in this case $e_1 - e_2$ becomes $(1, -1, 0, 0)$ (we ignore teh fifth coordinate), and we get

$$\begin{vmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}.$$

We can see this determinant is 1 by expanding by columns — we claim that we can always find a column with exactly 1 nonzero entry. In particular, this happens exactly when the corresponding vertex is a leaf.

It's an important property that every tree has at least two leaves. We always ignore the last vertex, so there should be at least one more vertex in the tree which is a leaf. That means the corresponding column only has one nonzero entry, and we can expand by that column which corresponds to removing the leaf. In the end, we will get ± 1 (by induction removing leaves).

Meanwhile, $(n-1)!\,\mathrm{Vol}\,P = 1$ requires that it is a simplex (if it's not a simplex, the volume is greater — 1 is the minimal possible value of this).

This gives us the following corollary:

Corollary 30.14

Let T be any directed tree on n vertices, and let $G = \overline{T}$. Suppose we then play the game with initial graph G. Let the endpoints of this game be $T^{(1)}, \ldots, T^{(N)}$, for some way of playing the game. Then $R_{\overline{T}}$ is the union o $R_{\overline{T^{(i)}}}$, and these pieces are simplices with non-overlapping interiors with volume $1/(n-1)!$. In particular,

$$(n-1)!\,\mathrm{Vol}\,R_{\overline{T}} = N.$$

Another observatino is that a tree equals its transitive closure if and only if it is alternating (by definition). So we have the transitive closure of some tree, and then the transitive closures of some other trees. So all the endpoints will correspond to trees, and therefore to alternating trees.

Why is this not a power of 2?

The game is a binary tree but at some points we'll get endpoints where we can't keep breaking them up.

Prof. Postnikov is tempted to call this a triangulation of the root polytope, but we have to be a little bit careful, and we should use the word *tesselation* instead. The difference is that in a triangulation, there is the extra property that for any two simplices, their intersection should be a common face. This may or may not be true for our subdivision — it's a subdivision into pieces of the same volume which don't overlap, but they may or may not properly intersect.

Example 30.15

Consider the case $n = 4$, where G is the transitive closure of the tree $1 \to 2 \to 3 \to 4$.

So in this case, $G = K_n^{\to}$.

This polytope lives in 4-dimensional space, but it's really 3-dimensional. We can't draw 3-dimensional pictures on the board, so we are going to represent it by a 2-dimensional picture instead.

To get this, we will have the origin, and we will have some vectors $e_1 - e_2$, $e_2 - e_3$, $e_3 - e_4$, and then $e_1 - e_3$ (which is the sum of our first two), $e_2 - e_4$ (the sum of the last two), and $e_1 - e_4$. Instead of drawing all of these things, we will draw a little affine hyperplane that goes close to but does not contain the origin, and we will intersect our polytope with this hyperplane. In particular, each of the rays from the origin to our vectors will intersect this hyperplane at a point. So we get a triangle corresponding to $e_1 - e_2$, $e_2 - e_3$, and $e_3 - e_4$; then we have one point on the midpoints of the first two and the last two, adn another point in the middle. So we get the following:

$\bullet\, e_2 - e_3$

$\bullet\, e_1 - e_3$ $\quad$ $\bullet\, e_2 - e_4$

$\bullet\, e_1 - e_4$

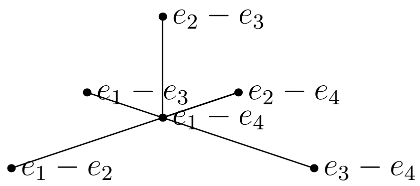
$\bullet\, e_1 - e_2$ $\qquad$ $\bullet\, e_3 - e_4$

(The middle thing should be on both the medians.)

One way to play the game is to replace our graph with the graphs $1 \to 2$, $1 \to 3$, $3 \to 4$ and another to $1 \to 3$, $2 \to 3$, $3 \to 4$. The transitive closure of the first graph should contain 12, 13, and 34; so we are cutting out the smaller triangles on the bottom and top.

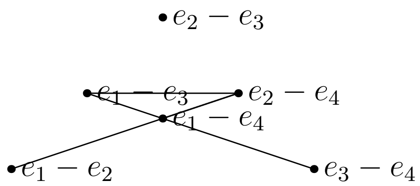
Our first piece then breaks into (picking $1 \to 3 \to 4$) $1 \to 2$, $1 \to 3$, $1 \to 4$ and $1 \to 2$, $1 \to 4$, $3 \to 4$. This divides our smaller triangle into two things.

This gives us the following thing:



These pieces all have the same volume (the picture looks misleading, but that's because we just took a 2d section). This particular subdivision corresponds to all non-crossing alternating trees.

But there is another way to play this game that will give us a different triangulation, corresponding to all non-nesting:



(Note: Prof. Postnikov is drawing the trees inside the triangle.)

Exercise 30.16. Find a bijection between non-crossing alternating trees, non-nesting alternating trees, and some other Catalan object.

§31 November 28, 2022

Recall that V_λ (for $\lambda \vdash n$) form the irreducible representations of S_n, and χ_λ are their characters. These are class functions, and these characters form a *basis* of the space of class functions.

Definition 31.1. When we talk about *characters* of S_n, these are exactly $\mathbb{Z}_{\geq 0}$-linear combinations of χ_λ. A **virtual character** is an arbitrary linear combination of χ_λ (not necessarily over $\mathbb{Z}_{\geq 0}$ — for example, over $\mathbb{C}$); these are the same as arbitrary class functions.

We previously discussed the Murnaghan–Nakayama rule for the values of characters (and more generally, the characters for skew shapes).

Today we will see another view on characters.

§31.1 Frobenius Character Map

Definition 31.2. The Frobenius character map is the function ch: $\{\text{class functions on } S_n\} \to \Lambda^n$ (where Λ^n is the space of homogenous symmetric functions in infinitely many variables of degree n), defined as follows: for any virtual character $\chi\colon S_n \to \mathbb{C}$,

$$\operatorname{ch}(\chi) = \frac{1}{n!}\sum_{w\in S_n} \chi(w)\cdot p_{\operatorname{type}(w)},$$

where type(w) is the cyclic type of w, written as a partition $\nu = (\nu_1, \ldots)$; and $p_\nu = p_{\nu_1}p_{\nu_2}\cdots$ is the product of power symmetric functions. (This is an isomorphism.)

We will use χ to denote characters, and ch to denote the Frobenius characteristic map.

We know that both sides have dimension the number of partitions, but this definition may seem a bit random — why power symmetric functions instead of elementary or monomial symmetric functions? But for reasons we'll see soon, this is the correct definition to make many things work nicely.

Here $\nu = (\nu_1, \ldots)$ is a partition of n, and

$$p_\nu = p_{\nu_1}\cdots,$$

and p_k is the power symmetric function $x_1^k + x_2^k + \cdots$.

Here we averaged $\chi(w)p_{\operatorname{type}(w)}$ over all permutations. But this is constant over conjugacy classes, so we can instead sum over all partitiosn of n: this gives

$$\operatorname{ch}(\chi) = \sum_{\nu\vdash n}\chi(\nu)\cdot\frac{p_\nu}{z_\nu},$$

where

$$\frac{1}{z_\nu} = \frac{1}{n!}\cdot\#\{w\in S_n \text{ of cyclic type } \nu\}.$$

In fact, there is an explicit formula for z_ν:

Lemma 31.3

$$z_\nu = \prod_{i\geq 1} i^{m_i}\cdot m_i!\,,$$

where m_i is the number of parts equal to i in ν.

Example 31.4

For the partition $\nu = (4, 2, 2, 1, 1, 1, 1)$, we have $m_1 = 4$, $m_2 = 2$, $m_3 = 0$, and $m_4 = 1$ (and all other m_i are 0). So

$$z_\nu = 4!\cdot(2^2\cdot 2!\,)\cdot(4^1\cdot 1!\,).$$

Proof. Fix a permutation $w \in S_n$ of cyclic type ν. Then by definition

$$\frac{n!}{z_\nu} := \#\{\text{permutations of cyclic type } \nu\} = |\operatorname{Orb}(w)|$$

where Orb(w) denotes the orbit of w in the S_n-action on S_n by conjugation — where a permutation u sends $w \mapsto uwu^{-1}$.

Now by a basic fact from group theory (true for any action of any finite group), we have

$$|\mathrm{Orb}(w)| = \frac{|S_n|}{|\mathrm{Fix}(w)|},$$

where $\mathrm{Fix}(w)$ consists of all permutations $u \in S_n$ that fix w under this action (or in other words, such that $uwu^{-1} = w$). So by definition, then

$$z_\nu = |\mathrm{Fix}(w)|.$$

This means for a given permutation w, we want to find the number of permutations u which fix w when acting by conjugation.

First let's describe what conjugation does. Suppose we have a permutation w, which we can write out in cycle notation; it'll have m_1 1-cycles, m_2 2-cycles, m_3 3-cycles, and so on (where all the points are labelled by numbers 1 to n). When we act by conuugation, we're keeping the same picture, but relabelling the vertices — permuting the labels according to u. So our goal is to look at this picture and see how many ways there are to relabel it such that the picture stays the same.

We can permute all the fixed points. For each of the 2-cycles, we can switch the labels on the 2-cycles, and also permute them. In general, for any cycle we can cyclically permute the entries in the cycle, and we can also permute the cycles themselves. This means

$$|\mathrm{Fix}(w)| = m_1! \cdot 2^{m_2} m_2! \cdot 3^{m_3} m_3! \cdots. \qquad \square$$

Now we'll see why the definition of the Frobenius character map is the correct one.

Theorem 31.5 (Frobenius)

We have $\mathrm{ch}(\chi_\lambda) = s_\lambda$. More generally, $\mathrm{ch}(\chi_{\lambda/\mu}) = s_{\lambda/\mu}$.

Recall that $s_{\lambda/\mu}$ denotes a skew Schur function.

Frobenius originally formulated this theorem in a different way:

Theorem 31.6

Suppose that $\lambda = (\lambda_1, \ldots, \lambda_k) \vdash n$, and $\rho = (k-1, k-2, \ldots, 1, 0)$. Then $\chi_\lambda(\mu)$ is the coefficient of the monomial $x^{\lambda+\rho}$ in

$$\prod_{1 \le i < j \le k} (x_i - x_j) \cdot p_\nu(x_1, \ldots, x_k).$$

Exercise 31.7 (Optional PS 3)**.** See why the two claims are equivalent.

Basically, they're equivalent where we define Schur functions classically (as a quotient of two determinants); if you unfold definitions carefully then they say the same thing.

§31.2 Logic of Proofs

There are three claims:

(1) The Murnaghan–Nakayama rule for $\chi_{\lambda/\mu}$.

(2) The Frobenius character formula.

(3) A version of the Murnaghan–Nakayama rule for Schur functions $s_{\lambda/\mu}$ — where instead of characters of irreducible representations, we talk about Scuhr functions.

We've already proven the first. Meanwhile, any two of these claims implies the third (if we know the MN rule and the Frobenius character formula, then that automatically converts it into a statement about Schur functions; conversely if we have the same rule for characters and Schur functions, then you know they're equal). If you are a representation theorist you may try to prove (1) and (2) to obtain (3); if you are a combinatorialist you may try to prove (1) and (3) to get (2).

Since we are combinatorialists, we will do the latter — we will show that the exact same Murnaghan–Nakayama rule holds if we replace characters by Schur functions.

§31.3 Murnaghan–Nakayama Rule for Symmetric Functions

Theorem 31.8

If we express a Schur function $s_{\lambda/\mu}$ as a sum of power symmetric functions, we get

$$s_{\lambda/\mu} = \sum_{\nu} \sum_{\text{ribbon tableau of shape } \lambda/\mu \text{ and type } \nu} (-1)^{\mathrm{ht}(RT)} \cdot \frac{p_\nu}{z_\nu}.$$

This ios what we'd get if we apply the Frobenius characteristic map to the first version. But we'll instead prove this theorem directly, and that'll automatically imply the Frobenius map.

So our goal is to express Schur functions in terms of power symmetric functions. But before we do this, we will prove another related result.

Theorem 31.9

We have

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

where the sum is over all $\lambda \supset \mu$ such that λ/μ is a k-ribbon.

As a corollary, we can let μ be empty:

Corollary 31.10

p_k is the alternating sum of Schur functions corresponding to hooks —

$$p_k = s_k - s_{(k-1,1)} + s_{(k-2,2)} - \cdots.$$

Example 31.11

We have

$$p_2 = s_{\ydiagram{2}} - s_{\ydiagram{1,1}} = h_2 - e_2.$$

If you take arbitrary monomials of degree 2 and subtract the squarefree ones, you get exactly the squares.

Example 31.12

We have

$$p_3 = s_{\ydiagram{3}} - s_{\ydiagram{2,1}} + s_{\ydiagram{1,1,1}},$$

which can also be checked by hand.

There are several approaches to proving this; we will use generating functions.

Proof. First, we can write the generating function for complete homogeneous symmetric functions as

$$H(t) = \sum_{k\geq 0} h_k t^k = \prod_{i\geq 1} \frac{1}{1 - x_i t}.$$

(If we expand as a geometric progression and open parentheses, we get the desired sum.) Similarly, we have

$$E(t) = \sum_{k\geq 0} e_k t^k = \prod_{i\geq 1} (1 + x_i t).$$

Finally, we can define a generating function for power symmetric functions, but in a slightly different way — we define

$$P(t) = \sum_{k\geq 1} p_k t^{k-1} = \sum_{i\geq 1} (x_i + x_i^2 t + x_i^3 t^2 + \cdots) = \sum_i \frac{x_i}{1 - x_i t}.$$

Now from these formulas, swe can notice a few relations between these generating functions:

- $H(t) \cdot E(-t) = 1$. (We've seen this before, and it's trivial from the first two equations.)
- $H'(t) = P(t)H(t)$ — we're taking our expression for H, taking tis derivative with respect to t, and using the product rule to write our answer as a sum over all i. Then we get back $H(t)$ multiplied by this expression.
- $E'(t) = P(-t) \cdot E(t)$.

Remark 31.13. 2 and 3 are known as *Newton's formulas*. If we expand this, then we get a rule to recursively express the power sums in terms of complete homogeneous symmetric functions; similarly we can express them in terms of elementary symmetric functions.

There is another way to express power symmetric functions in terms of h's and e's. Using the second formula, we can write

$$P(t) = \frac{H'(t)}{H(t)} = H'(t) \cdot E(-t).$$

This gives us a way of expressing power sum sin terms of complete homogeneous and elementary symmetric functions. More explicitly, we get the following lemma (by extracting the kth coefficient):

Lemma 31.14

$$p_k = \sum_{r=1}^{k} r h_r \cdot (-1)^{k-r} e_{k-r}.$$

But we know the rule for how to multiply Schur functions by h_i and e_j (the Pieri formulas). So we want to get our formula for ribbons as a corollary of the Pieri formulas.

We can write

$$p_k s_\mu = \sum_{r \geq 1, \ell \geq 0, r+\ell=k} r(-1)^\ell e_\ell h_r \cdot s_\mu.$$

Now for $(h_r s_\mu)$, we have the Pieri formula — this is done by adding a horizontal r-strip. Meanwhile, multiplying by e_ℓ corresponds to adding a vertical ℓ-strip.

So we start with some μ, and then look at all possible ways to add a horizontal r-strip, and after that we add a vertical ℓ-strip. But there is an extra factor of r (the red boxes), so we want to also put a dot in one of the red boxes. And this diagram counts with a sign $(-1)^\ell$ (where ℓ is the number of blue boxes).

As we see, this shape looks a bit like a ribbon, but it's not necessarily connected, and one red box is marked. The idea is to take this set and cancel all unnecessary terms (with a sign-reversing involution) so that only connected shapes remain, and there's only one position for a dot. □

§32 November 30, 2022

Last time, we began proving the following theorme:

Theorem 32.1

$p_k s_\mu = \sum_\lambda (-1)^{\mathrm{ht}(\lambda/\mu)} s_\lambda$, where the sum is over all λ obtained from μ by adding a k-ribbon.

In particular, when μ is empty, we get an expansion of p_k based on hooks:

Corollary 32.2

$p_k = s_{(k-1,1)} + s_{(k-2,1)} + \cdots$.

This rule also tells us how to multiply s_λ by a product of power sums (since we add a ribbon, and then add another ribbon, and so on):

Corollary 32.3

FOr any partition $\nu = (\nu_1, \ldots, \nu_\ell)$, we have

$$p_\nu s_\mu = \sum_{\lambda \subseteq \mu} \sum_{\text{RT ribbon tableau of shape } \lambda/\mu,\text{ type } \nu} (-1)^{\mathrm{height(RT)}} s_\lambda.$$

THis should kind of remind us of the other formula formulated last time: denote this coefficient we have here by $\chi_{\lambda/\mu,\nu}$. This is exactly the value of the character (by the Murnaghan–Nakayama rule), but right now we'll use this to just denote the combinatorial expression.

Now if we specialize to the case where μ is empty, this says

$$p_\nu = \sum_\lambda \chi_{\lambda,\nu} s_\nu.$$

So $\chi_{\lambda,\nu}$ are the coefficients if we want to expand the power symmetic functions (and their products) in terms of the Schur symmetric functions.

But we actually mentioned a slightly different thing, on how to express Schur functions in terms of power sums. The formula is actually quite similar —

$$s_\lambda = \sum_\nu \chi_{\lambda,\nu} \frac{p_\nu}{z_\nu},$$

where $z_\nu = \prod_i i^{m_i} m_i!$, where m_i is the number of i's in ν.

(Here ν is a partition.)

These formulas look similar, but they're nto exaclty the same. So the first question is, how do we see the relationship between them?

First let's prove our theorem.

Proof. First, we used generation functions to show that the power symmetric functions can be expressed in terms of the elementary and complete homogeneous functions, as

$$p_k = \sum_{i=1}^k r \cdot h_r \cdot (-1)^{k-r} e_{k-r}.$$

Meanwhile, the Pieri rules let us multiply Schur functions by complete homogeneous and elementary symmetric functions —

$$h_r s_\mu = \sum_{\lambda/\mu \text{ is a horizontal } k\text{-strip}} s_\lambda,$$

and

$$e_\ell s_\mu = \sum_{\lambda/\mu \text{ is a vertical } \ell\text{-strip}} s_\lambda.$$

We can now combine these formulas, which tells us that

$$\begin{aligned} p_k &= \sum_{r \geq 1, \ell \geq 0, r+\ell=k} r(-1)^\ell e_\ell h_r s_\mu \\ &= \sum_\lambda \sum_{\lambda \supseteq \gamma \supsetneq \mu} r(-1)^\ell s_\lambda. \end{aligned}$$

Here we're adding a horizontal r-strip and a vertical ℓ-strip. (Note that r can't be 0, but ℓ can.) THe sum over $\lambda \supseteq \gamma \supsetneq \mu$ is over stuff such that λ/γ is a vertical ℓ-strip, and γ/ν a horizontal r-strip.

Now we have this expression in terms of s_λ, and we want to understand what it is.

We can imagine startning out with any μ. Then we add some horizontal strip, which we are drawing in red (we are usign blue for vertical because of the sky). And then we add a vertical strip, which we are drawin gin blue. So we get a diagram that looks like htis. But we have an extra factor of r, which emans every diagram ilke this comes with a coefficient the number of red boxes; so we do this by marking a red box. And the other thing is that we have a sign $\pm$, whcih is (-1) to the number of blue boxes.

Now we have these diagrams, which kind of look a bit like ribbons, but there is a difference — by definition ribbons have to be connected, and these shapes are not necessarily connected (and they are also colored and marked).

Example 32.4

Consider $k = 2$. Then there are several options: first, we should have at least one red box (but we might have zero blue boxes). So one possibility is two red boxes with the left one marked

$$\begin{ytableau} \bullet \end{ytableau}$$

or with the right one marked, or with two disconnected red boxes with either one marked.

We can also have one red box and one blue box — horizontal domino, vrtical domino, or two disconnected ones with either one on the top-left (nad with the red one marked).

The top guys come with a $+$, and the bottom guys come with $-$.

Now the idea is to construct a sign-reversing involution on these colored diagrams. If we forget about colors and markings, this thing should preserve the shape; but we want to reverse the sign.

One way to do this is to switch the color of one box. In the cases with just one blue box, we can see that if we switch the color of the thing, then we sometimes get stuff. But in the vertical domino, if we switch the color of the bottom blue box, this is bad — we cannot perform the operation because we get a thing that's not horizontal. But in the other cases we again switch color to get an appropriate diagram.

The involution is also not defined for the horizontal domino with a marked box on the right (both red) — we cannot switch the color of a marked box (which must be red), and if we switch the red one to blue then we get an invalid thing.

So we can cancel everything else, and we are left with the contribution of these two diagrams. So we have the two situations

$$\ydiagram{2} - \ydiagram{1,1},$$

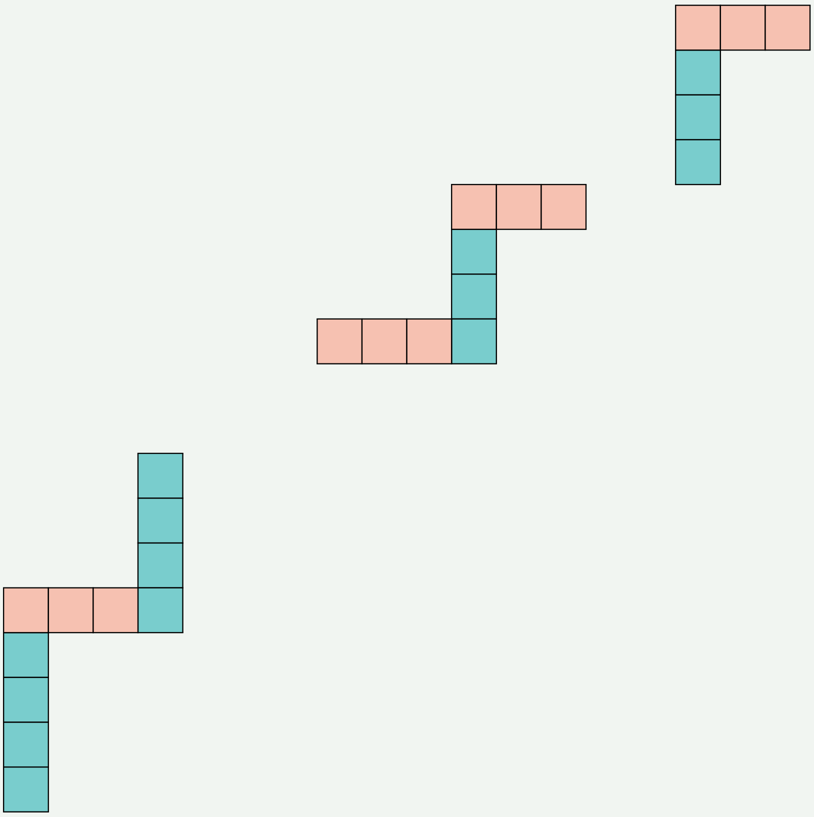
which exactlyn corresponds to our two ribbons.

Now let's do this in general. We have a sign-reversing involution on colored and marked diagrams, defined by the following rule:

- Find the top-rightmost box such that:
  - it is unmarked, and
  - It is t he top-rightmost box in its connected component.
- Switch its color.

Example 32.5

In the case



we look at the top-right box in each component. We first look at the furthest component; if it is marke,d then we go to the next. But if we had a mark anywhere else, we would use this thing.

It's clear if you repeat this thing twic,e you get back to the original; so this is really an involution. I tis also clar that it changes the number of blue boxes by 1. This involution is not defined when the diagram has one connected component, and its top-rightmost box is marked.

So we can cancel out everything except the ones where the thing isnt' defined. But these are in bijection with ribbons:

Claim — Any uncolored and unmarked ribbon can be uniquely colored and marked like this.

Example 32.6

Suppose you give me any unmarked uncolored ribbon $rrruurrrrruurrr$. Then the last box has to be marked red, so everything to its left has to be red; then we go left one step adn that stuff has to be red, and so on. And then the vertical portions have to be blue. (The thing can't ve blue because it would cause two red boxes in the same column.)

And also the sign is (-1) to the number of blue boxes; this is exactly (-1) to the height of λ/μ. (The number of blue boxes i sht enumber of rows minus one — the top row will be all red, and in all ohter rows you'll have one blue box.)

Student Question. Would you get something symmetric if you put vertical strips before horizontal strips?

Height is related to width and to total number of boxes, so the height sign and width sign are related. If you first multiply by e_ℓ nad h_r you will get the same result, but the diagram will be slightly different. □

So what we have proved is how to multiply, in generla, any product of p_k by any s_ν, and the result si what we denoted by $\sum \chi_{\lambda/\mu,\nu} s_\lambda$. But this is not exactly what we called the symmetric function version of the MN rule. What we wanted was how to expand a Schur f unction interms of power sums:

Lemma 32.7

$s_{\lambda/\mu} = \sum_\nu \chi_{\lambda/\mu,\nu} \frac{p_\nu}{z_\nu}$.

So we want to understnad is how to relate these two formulas — is there a way to deduce one formula from the other?

To understand this, we want to look at Hall's inner product of symmetric functions.

§32.1 Hall's Inner Product of Symmetric Functions

Theorem 32.8

There exists a unique inner product $\langle -, - \rangle$ on $\Lambda_{\mathbb{R}}$ (this can be done over complex numbers as well, but the field should have characteristic 0) — the space of symmetric functions with real coefficients, such that:

1. $\langle s_\lambda, s_\mu \rangle = \delta_{\lambda\mu}$ (i.e. 1 if $\lambda = \mu$ and 0 0therwose). Alternatively, we already know the Shcur functions form a basis for Λ_n; but here we are requiring them to be an orthonormal basis. [Of course this satisfies the uniqueness claim, so you can use this as the definition of the inner product — there is auique such thing.)
2. For two partitions, $\langle p_\lambda, p_\nu \rangle = z_\lambda \delta_{\lambda,\mu}$. In other words, the P_λ form an orthogonal (but not orthonormal) basis; to normalize you'd divide by $\sqrt{z_\lambda}$. (This is z_λ if $\lambda = \mu$ and 0 otherwise, so it is symmetric.)
3. $\langle h_\lambda, m_\mu \rangle = \delta_{\lambda\mu}$ Recall that htese are complete homogeneous adn monomial. In other words, the basis of h's is dual to teh basis of m"s.
4. You may be wondering what happens if you take the elementary. This is dual to another type of symmetrc functions we have not discussed.
5. $\langle s_\lambda, f s_\mu \rangle \langle s\lambda/\mu, f \rangle$ for every $f \in \Lambda$ which is a class function.

This theorem implies our two identities rae equivalent. To see why, (*) says that our coefficients $\chi_{\lambda/\mu,\nu}$ are the inner product $\langle p_\nu s_\nu s_\nu, s_\lambda \rangle$ and the secondi= condition becomes that ew should have $\chi_{\lambda/\mu,\nu} = s_{\lambda/\nu} p_\nu$. The equality between these thungs folows from Condition (4) —- so the theorem immediately implies our two things are equivalent.

Let's suppose we already have this inner product (we can define it by the first condtion). We'll now look at some examples of the last condition:

Example 32.9

When $f_1 = s_1 = x_1 + x_2 + \cdots$, we are saying that $\langle s_\lambda, s_\mu s_1 \rangle$ is 1 if $\lambda \cdot\!> \mu$ in $\mathbb{Y}$ and 0 otherwise.

Example 32.10

Suppose $\mu$is a single box and $r = s_\nu$. The conditino tells us

$$\langle s_\lambda, s_1 s_\nu \rangle$$

The left-hand side is 1 if $\lambda \cdot\!> \mu$ and 0 otherwise.

So this tells us if we take λ and minus a single box, then we get he sum of μ covered by λ of s_μ. In other words, if we remove an upper-efbox and look at our Schur function, that is the same as removing any of the corner boxes.

We an ask combinatorially there should be a bijection betwen SSYT of the sligtly skew shpaena nd straight shapes obtained gy λ by removing one of the corner boxes. This is a thing that exists; it is like a game of fifteen where you start sliding boxes, and it's related to toggles and RSK, so we will talk about this stuff on Friday.

§33 December 2, 2022

Definition 33.1. The **Hall inner product** $\langle -, - \rangle$ on the set Λ of symmetric functions over $\mathbb{R}$ is defined by

$$\langle s_\lambda, s_\mu \rangle = \delta_{\lambda\mu}$$

(1 if $\lambda = \mu$ and 0 otherwise).

Proposition 33.2 (Cauchy-Like Identities)

We can write

$$\prod_{i,j} \frac{1}{1 - x_i y_j} = \sum_\lambda s_\lambda(x) s_\lambda(y) = \sum_\lambda m_\lambda(x) \cdot h_\lambda(y) = \sum_\lambda \frac{p_\lambda(x) p_\lambda(y)}{z_\lambda}.$$

The first is the usual Cauchy identity, and it can be proved for example by RSK. For the second identity, this is actually much easier, and it basically follows from the definition — we can write out

$$\prod \frac{1}{1 - x_i x_j}$$

and attempt to find the coefficient of $m_\lambda(x)$ in this infinite product. To find the coefficient of $m_\lambda(x)$, it's enough to find the coefficient of one monomial, say $x_1^{\lambda_1} x_2^{\lambda_2} \cdots$. And we can explicitly extract — it's clear that the coefficient of $x_1^{\lambda_1}$ in all the terms with x_1 is $h_{\lambda_1}(y)$, and the next is $h_{\lambda_2}(y)$, and so on. The third point is also fairly easy, from the definition of power sums.

Proposition 33.3

Two linear bases $\{u_\lambda\}$ and $\{v_\lambda\}$ of Λ are dual to each other with respect to the Hall's inner product if and only if

$$\sum_\lambda u_\lambda(x) v_\lambda(y) = \prod \frac{1}{1 - x_i y_j}.$$

This is pretty easy to check from the definitions —

Proof. We can expand out u_λ in terms of Schur functions, as $u_\lambda = \sum_\nu a_{\lambda\nu} s_\nu$ — so we have some coefficients forming a matrix $A = (a_{\lambda\mu})$. (Thsi is an $\infty \times \infty$ mtarix; if you don't like infinite matrices, you can always restrict to partitions of a given size.) We can similarly expand $v_\mu = \sum_\gamma b_{\mu\gamma} s_\gamma$, giving a matrix $B = (b_{\mu\gamma})$.

The bases u and v are dual to each other if and only if (by definition) $\langle u_\lambda, v_\nu \rangle = \delta_{\lambda\mu}$. We can now plug in these expressions to get

$$\langle \sum_\nu a_{\lambda\nu} s_\nu, \sum_\gamma b_{\mu\gamma} s_\gamma \rangle = \delta_{\lambda\mu}.$$

This is equivalent to saying that $\sum_\nu a_\lambda \gamma b_{\mu\nu} = \delta_{\lambda\mu}$, or equivalently that $AB^\intercal = \mathrm{Id}$, or that $A = (B^\intercal)^{-1}$.

On the other hand, we can also rewrite our Cauchy-type identity in this form — it's equivalent to saying that

$$\sum_\lambda u_\lambda(x) v_\lambda(y) = \sum_\lambda s_\lambda(x) s_\lambda(y)$$

(by the Cauchy identity for Schur functions). This gives

$$\sum_\lambda \sum_{\nu,\gamma} a_{\lambda\nu} s_\nu(x) b_{\lambda\gamma} s_\gamma(y) = \sum s_\lambda(x) s_\lambda(y).$$

So this means

$$\sum_\lambda a_{\lambda\nu} b_{\lambda\gamma} = \delta_{\nu,\gamma}.$$

This means $A^\intercal B = \mathrm{Id}$, or equivalently $A = (B^{-1})^\intercal$. The key property is that inverse commutes with transpose, so these are equal. $\square$

What we did here is that if you consider this thing as a symmetric function in x, then the coefficient of $s_\nu(x)$ should be $s_\nu(y)$.

As a corollary, we have proved that:

Theorem 33.4

By definition $\langle s_\lambda, s_\nu \rangle = \delta_{\lambda\mu}$; also $\langle h_\lambda, m_\mu \rangle = \delta_{\lambda\mu}$, and $\langle p_\lambda, p_\mu \rangle = z_\lambda \delta_{\lambda\mu}$.

So h and m are dual, and p is dual to a rescaled version.

Theorem 33.5

For every $f \in \Lambda$, we have

$$\langle s_\lambda, s_\mu f \rangle = \langle s_{\lambda/\mu}, f \rangle.$$

In particular, if f is itself a Schur function $f = s_\nu$, then we get

$$\langle s_\lambda, s_\mu s_\nu \rangle = \langle s_{\lambda/\mu}, s_\nu \rangle.$$

But in fact this is not just a special case; it implies the left identity, because the Schur functions form a basis, so if the identity holds for every basis element then it holds for everything.

These inner products have special names:

Definition 33.6. These are called the **Littlewood–Richardson coefficients**.

There are two ways to think about them — expanding $s_\mu s_\nu$ in the basis of Schur functions, or expanding a skew Schur function in the basis of (non-skew) Schur functions.

Proof. It's enough to prove this for any linear basis for Λ. In the example we took Schur functions, but we can take another; so it is enough to prove this for f being complete homogeneous functions $h_\nu = h_{\nu_1} h_{\nu_2} \cdots$. (These form a basis by the fundamental theorme of symmetric functions.)

Now notice that $s_\mu h_\nu = s_\mu h_{\nu_1} h_{\nu_2} \cdots$. The Pieri formulas tell us how to multiply a Schur function by one h_{ν_1} — this is given by adding a horizontal strip. So then we add a ν_1-strip and then ν_2-strip and so on, which means we get the sum over all λ containign μ of s_λ times the number of SSYT of shape (λ/μ) and weight ν.

This means the left-hand side $\langle s_\lambda, s_\mu h_\nu \rangle$ is exactly the coefficient of s_λ here, so it si the number of SSYT of skew shape λ/μ and weight ν.

On the other hand, on the right-hand side $\langle s_{\lambda/\mu}, h_\nu \rangle$, we know h is dual to the basis of monomials, so this is equivalently the coefficient of m_ν in $s_{\lambda/\mu}$. By the *definition* of Schur functions, this is the number of SSYT of shape λ/μ and shape ν. ∎

Student Question. How do you realize the correct basis to think of for f is the h_λ?

We know how to expand s_λ/μ in terms of monomial functions, and expansion in terms of monomials is equivalent to calculating the inner product with the dual basis, and m is dual to h.

§33.1 MN Rule

Last time we saw two formulas:

Theorem 33.7

$p_\nu s_\mu = \sum_\lambda \chi_{\lambda/\mu,\nu} s_\lambda$.

$s_{\lambda/\mu} = \sum_\nu \chi_{\lambda/\mu,\nu} p_\nu / z_\nu$.

Last time we wanted to see how these two formulas are equivalent.

The first formula says that

$$\chi_{\lambda/\mu,\nu} = \langle p_\nu \cdot s_\mu, s_\lambda \rangle.$$

The second says that

$$\chi_{\lambda/\mu,\nu} = \langle s_{\lambda/\mu}, p_\nu \rangle.$$

The above theorem implies that these two expressions are equal, so the two statements are equivalent.

§33.2 Special Case

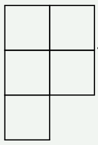
Now suppose that μ consists of a single box and $f = s_\nu$. Then we have

$$\langle s_\lambda, s_1 s_\nu \rangle = \langle s_{\lambda/\square}, s_\nu \rangle.$$

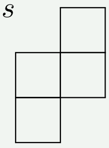
On the other hand, we know how to multiply Schur functions by s_1, so this is the sum over all μ obtained by adding a box to ν of s_ν. In particular, this inner product is equal to 1 if $\lambda \gtrdot \nu$ in Young's lattice and 0 otherwise. This should be the same as the right-hand side, so we have deduced that $s_{\lambda/\square} = \sum_{\nu \lessdot \lambda} s_\nu$.

Example 33.8

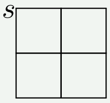
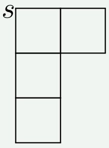
Suppose we take the straight shape

$\lambda =$  .

Then we are looking at

s  .

The claim is that this is the same as

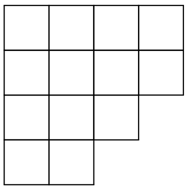
s  $+\, s$  .

Corollary 33.9

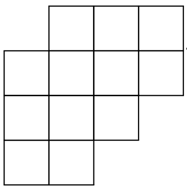
For any λ a partition and β a composition, the number of SSYT of shape $\lambda/\square$ and weight β is equal to the sum over all μ obtained by removing a box from λ of the number of SSYT of shape ν and weight β, i.e. $\sum_{\nu \lessdot \lambda} \#\mathrm{SSYT}(\nu, \beta)$.

We can ask: is there a nice bijection that proves this identity? THe answer is yes.

Take the example



and remove the first corner to get

 .

Suppose we then make this a SSYT by putting numbers in the boxes, as

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

We want a correspondence between SSYT and SSYT of similar shape, but where we remove a corner box instead. This bijection is called **Jeu de toqin**, or 'game of 15'. We can think of this as a shape with one unfilled box, and we want to start sliding entries around such that at every step we get a semistandard Young tableau (for example we cannot slide the top row to the left because we would break).

So for example, we can slide the 1 up to

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

.

We cannot slide 4 here, but we can slide 2 here, so then we get

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

.

Essentially, there is always exactly one way to slide — if one is bigger then we need to slide the smaller one, if they're equal we slide from the bottom. So we then slide 2, then 5, and we end up with one empty outer box

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

.

These operations are invertible — if we have a diagram with an empty corner box, then there is a unique way to slide things back into that box.

An even more general fact is true:

Fact 33.10 — Suppose we have any skew shape λ/μ, and k a nonnegitive number. If we look at all possible ways gto remove a horizontal strip from the upper-left side of the shape and we take the Schur function of the remaining thing, this is equal to the sum over lal ways to remove a k-strip on the other side.

(Above is gthe case where $k = 1$ and we have a skew shape.)

Question 33.11. Is there a bijective proof?

We again want a correspondence — fix some weight and fill in the boxes leaving certain things empty.

Suppose we fix λ/μ and some composition $\beta = (\beta_1, \ldots)$, and then we look at the number of SSYT of shape λ/μ and weight $(k, \beta_1, \beta_2, \ldots)$ — since we can think of these as putting 0's in all the empty boxes. On the other hand, the thing on the right is the number of SSYT with the same shape and weight $(\beta_1, \beta_2, \ldots, \beta_\ell, k)$. So basically, we want a correspondence between these two things.

We already have this correspondence from the beginning of the semester — when we discussed the Bender–Knuth involutions and toggles. This gives a way to permute two adjacent entries in the weight vector, and we can repeat this operation.

On the other hand, there is another construction, based on jeu de toqin — we can do the same thing.

We take this SSYT with the shape and with the shaded boxes empty. We then first look at the rightmost box of the first strip, and we perform slides to slide something into there. Then after that, we do another operation for the next box, and so on. So we get k paths. One can show that these paths are non-crossing (if you perform your operations from right to left). So then the empty boxes we leave form a horizontal k-strip.

Question 33.12. What is the relationship between the BK operations and this?

The elementary building blocks have changed from togglign to sliding, so what's the relationship between toggles and slides?

§33.3 Littlewood–Richardson Coefficients

Definition 33.13. $c^\lambda_{\mu\nu}$ is defined either as $\langle s_\lambda, s_\mu s_\nu\rangle$ (i.e. the coefficient of s_λ in $s_\mu s_\nu$ when we expand with respect to Schur functions), or as $\langle s_{\lambda/\mu}, s_\nu\rangle$ (i.e. we expand skew schur functions with respect to normal ones).

The LR rule gives us a combinatorial rule for these numbers. We will discuss many differenet constructions for these things. One thing is actually there is one obvious symmetry — you can switch μ and ν, because $s_\mu s_\nu = s_\nu s_\mu$. But there are also other symmetries, and ideally one would like to find a rule that makes these symmetries clear. The puzzle rule kind of gets very close, but it is still an open problem to find a Littlewood–Richardson rule that has all these symmetries. (In fact this is the most non-obvious symmetry.)

§34 December 5, 2022

§34.1 Littlewood–Richardson Coefficients

There are two ways to define the LR coefficients — $c^\lambda_{\mu\nu} = \langle s_\lambda, s_\mu s_\nu\rangle = \langle s_{\lambda/\mu}, s_\nu\rangle$. In the first definition, we're thinking about how to expand a product — $s_\mu s_\lambda = \sum_\lambda c^\lambda_{\mu\nu} s_\lambda$ — and in the other we're thinking about how to expand a *skew* Schur function — $s_{\lambda/\mu} = \sum_\nu c^\lambda_{\mu\nu} s_\nu$.

Today we'll find a combinatorial rule for these coefficients. There's a classical combinatorial rule, but these coefficients have lots of interesting symmetries and nice properties, which are not always obvious from the classical construction. So people found other constructions that make these properties more obvious. For that reason, we'll see the classical rule and several generalizations or different constructions.

First, there are several ways to think about these coefficients algebraically and geometrically, which each may make more sense in different points of view.

First, we can think about representations of S_n. We've seen that irreducible representations of S_n are labelled by partitions — and the Frobenius character makes $V_\lambda \rightsquigarrow s_\lambda$. On the right-hand side when we take the product, we'd think we'd get the tensor product, but this isn't true. Instead we get a different operation on representations (when we talk about representations $|\lambda| = n$, and when we multiply things, the number of boxes increases — so we are working not with one symmetric group but with all of them together). So instead the operation we need is a different one — when we have two irreducible representations V_μ and V_γ of S_m and S_n respectively, we define

$$V_\mu \circ V_\nu = \mathrm{Ind}_{S_m \times S_n}^{S_{m+n}} V_\nu \otimes V_\mu$$

(i.e. we take the tensor product, and induce this to S_{m+n}). The coefficients correspond to the expansion of this representation in the basis of irreducible representations — it expands as

$$\bigoplus_{\lambda \vdash m+n} V_\lambda^{c^\lambda_{\mu\nu}}.$$

> **Remark 34.1.** Taking the usual tensor products, working with a single symmetric group, will produce different coefficients, called the *Kronecker coefficients*. These are actually much harder to calculate — there are lots of nice combinatorial rules for the LR coefficients, but we don't know many things for the Kronecker coefficients except in special cases.

The second way of thinking about these coefficients is from the representations of GL_n. In this case, we *fix* n. Now irreducible representations of GL_n are labelled by things pretty close to partitions (but not the same) — we will use $V(\lambda)$ to denote irreducible representations of $\mathrm{GL}(n)$, where $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_n)$ is a sequence of weakly decreasing integers (which are allowed to be negative). So these are sort of like partitions, but with fixed number of parts and with negative parts allowed.

If we then take the usual character of $V(\lambda)$, then we get the Schur function

$$\chi_{V(\lambda)} = s_\lambda(x_1, \dots, x_n).$$

This is slightly more general than a polynomial — first $\lambda + a(1, 1, 1, \dots, 1)$ corresponds to $(x_1 \cdots x_n)^a \cdot s_\lambda$, so we can reduce to the case where all parts are nonnegative. But sometimes it's more convenient to allow negative parts. (So essentially this is a Schur function, but if λ has negative parts then we handle it by dividing by some power of the x's.)

(We can define the character on *diagonal* matrices, and then extend to arbitrary matrices using the fact that the character is a class function. The function defined on diagonal matrices is given by this Schur function.)

In this construction, then the usual tensor product is

$$V(\mu) \otimes V(\nu) = \bigoplus_\lambda V(\lambda)^{c^\lambda_{\mu\nu}}.$$

The next point of view is the geometry of Grassmanians. We now fix two numbers $0 \le k \le n$. Then Gr_{kn} is defined as the variety (or manifold) of the k-dimensional subspaces in $\mathbb{C}^n$. This is a nice variety that has a nontrivial topology, and an interesting cohomology group. Its cohomology group $H^*(\mathrm{Gr}_{kn})$ is closely related to symmetric functions — it is the ring of symmetric functions, but truncated, so we take Λ and quotient out by the linear span of s_λ for all λ which do not fit into a $k \times (n-k)$ rectangle. (Essentially we are killing all Schur functions which don't fit into this rectangle.)

From a geometric point of view, this ring has a nice basis, called the basis of Schubert classes. This basis $[x_\lambda] \mapsto s_\lambda$ for $\lambda \subseteq k \times (n-k)$. In particular, the product of Schur functions corresponds to taking a product of Schubert varieties, which has a geometric interpretation. There's a fact called Poincare duality on this cohomology ring — x_λ is dual to $x_{\lambda^\vee}$, where $\lambda^\vee$ is obtained by taking the complement of λ in the $k \times (n-k)$ rectangle and rotating it — $\lambda^\vee = (n-k-\lambda_k, n-k-\lambda_{k-1}, \dots)$.

Suppose we define $\tilde{c}_{\lambda,\mu,\nu} = c^{\lambda^\wedge}_{\mu\nu}$. This geometrically is the **intersection number** of Schubert varieties $[x_\lambda]$, $[x_\mu]$ and $[x_\nu]$. (This relabelling depends on n and k, but interestingly it is symmetric under permuting λ, μ, and ν.)

§34.2 Some Properties of the Littlewood–Richardson Coefficients

We'll now list some properties of the coefficients:

- $c^\lambda_{\mu\nu} \in \mathbb{Z}_{\ge 0}$.
- The involution $\omega\colon s_\lambda \to s_{\lambda'}$ (the transpose of λ, not the operation above) preserves the coefficients — $c^{\lambda'}_{\mu'\nu'} = c^\lambda_{\mu\nu}$.
- S_3-symmetry — suppose we fix n and k and assume all partitions fit into a $k \times (n-k)$ rectangle, and define $\tilde{c}_{\lambda\mu\nu}$ as above. Then $\tilde{c}_{\lambda\mu\nu}$ is symmetric.

It's also possible to get a S_3-symmetric version of the LR coefficients by using representations of GL_n — suppose we fix n (but not k), and require that $\lambda_1 = (\lambda_1, \ldots, \lambda_n)$ and so on (parts may be negative). There's an operation on representations sending $\lambda \to \lambda^* = (-\lambda_n, -\lambda_{n-1}, \ldots, -\lambda_1)$. Then we can define $\tilde{\tilde{c}}_{\lambda,\mu,\nu} = c^{\lambda^*}_{\mu\nu}$. This will also be symmetric under permutations of λ, μ, and ν. (In this interpretation λ, μ, and ν are slightly different things than partitions.)

We would now like to find a rule that explains at least some of these properties.

§34.3 Classical Littlewood–Richardson Rule

We'll think of the Littlewood–Richardson coefficients in terms of expanding the *skew* Schur functions — $s_{\lambda/\mu} = \sum_\nu c^\lambda_{\mu\nu} s_\nu$.

We know how to expand in terms of *monomials*, with the coefficients given by the Kostke numbers —

$$s_{\lambda/\mu} = \sum_\nu K(\lambda/\mu, \nu) m_\nu.$$

The Kostke numbers $\mathrm{SSYT}(\lambda/\mu, \nu)$ by definition count the number of SSYT of shape λ/μ and type ν.

We'll now try to count some special types of SSYT.

Definition 34.2. A **Yamanouchi word** (a.k.a. lattice word or ballot word) is a sequence $w = (w_1, \ldots, w_N)$ of strictly positive integers (not necessarily a permutation) such that for every initial subword (e.g. w_1, w_2, …, w_k),

$$\#1's \geq \#2's \geq \#3's \geq \cdots.$$

Remark 34.3. The reason they're called ballot words is that you can imagine there are candidates labelled by positive integers, and the first person casts the vote for the w_1st candidate, and so on. Then the ballot problem is to find the probability that the first candidate was always leading.

Example 34.4

The sequence 1, 1, 1, 2, 1, 3, 2 is a Yamanouchi word. On the other hand, 1, 1, 1, 2, 1, 3, 3 is bad because at this point we have two 3's and only one 2.

Exercise 34.5. What is the number of Yamanouchi words of length $2n$ with n 1's and n 2's?

The answer is C_n — thinking of 1 as an up-step and 2 as a down-step, these words become exactly Dyck paths. So we can think in general of Yamanouchi words as some sort of multi-dimensional Dyck path.

Definition 34.6. For a SSYT T, **Hebrew reading word** of T is obtained by reading the entries of T by rows from right to left, and top to bottom.

Remark 34.7. In any Semitic language, you read from right to left; this is where the name comes from, and it may also be called a Semitic reading word or Arabic reading word.

Example 34.8

Suppose we have the skew shape

$$T = \begin{array}{ccccc} & & 1 & 1 & 2 \\ 1 & 2 & 2 & 3 & 4 \\ 3 & 4 & & & \end{array}.$$

The corresponding words is 2, 1, 1, 4, 3, 2, 2, 1, 4, 3.

Remark 34.9. Some authors read in the opposite way — from left to right but bottom to top.

Definition 34.10. A Littlewood–Richardson tableau is a SSYT such that its Hebrew reading word is a Yamanouchi word.

Now we're ready to formulate the classical formulation of the rule:

Theorem 34.11

$c^{\lambda}_{\mu\nu}$ is the number of LR-tableaux of shape λ/μ and weight ν.

Remark 34.12. There is a long story of this theorem — it was formulated by Littlewood–Richardson in the 1930s, then Richardson proved it but with some gaps. The first proof was given in the 1970s. Later people have found a few short proofs that can fit into one lecture, so next lecture we will see one such proof.

There are a few other variants or generalizations of this rule.

§34.3.1 Zelevinsky's Pictures

Suppose we have *two* skew shapes λ/μ and ν/γ, and we want to calculate $\langle s_{\lambda/\mu}, s_{\nu/\gamma}\rangle$. Then the LR rule generalizes in two different ways. If we assume that γ is empty, then we get the Littlewood–Richardson coefficients — $\langle s_{\lambda/\mu}, s_{\nu}\rangle$. On the other hand, we can also get a rule for $\langle s_{\lambda}, s_{\mu} \cdot s_{\nu}\rangle$ — this is also a special case where μ is empty and the skew shape ν/γ is obtained by taking two regular shapes μ and ν and gluing them together. (Not the same μ and ν.) So these numbers specialize to the usual LR coefficients in at least two different ways.

One would like to find a combinatorial rule. But here there's additional symmetry — there's a symmetry between switching λ/μ and ν/γ.

(These coefficients are nonzero only when the two skew shapes have the same number of boxes.)

Definition 34.13. A **Zelevinsky picture** is a bijection φ between boxes in λ/μ and ν/γ such that:

(1) The Hebrew reading word of the boxes of λ/μ maps into a SYT of shape ν/γ — if we start reading the boxes of λ/μ from right to left and we see where these boxes go, on the right-hand side we get a SYT.

(2) The same condition for the inverse of the map — if we start listing boxes of the second shape by rows from right to left, this corresponds to some listing of boxes of the first shape.

It is quite nontrivial to find an example of such a correspondence.

Example 34.14

Suppose our shapes are

$$\lambda/\mu = \ydiagram{1+2,3,1} \text{ and } \nu/\gamma = \ydiagram{1+3,3}.$$

In this case, there is only one Zelevinsky picture:

$$\begin{ytableau} \none & 2 & 1 \\ 5 & 4 & 3 \\ 6 \end{ytableau} \to \begin{ytableau} \none & 1 & 2 & 5 \\ 3 & 4 & 6 \end{ytableau}.$$

Meanwhile, if we list the boxes of the right tableaux as

$$\begin{ytableau} \none & 3 & 2 & 1 \\ 6 & 5 & 4 \end{ytableau},$$

then we end up with

$$\begin{ytableau} \none & 2 & 3 \\ 1 & 5 & 6 \\ 4 \end{ytableau},$$

which is also a SYT.

Exercise 34.15. Check that these conditions are equivalent to the Littlewood–Richardson rule under both of the two specializations (the one where γ is empty, and the one where the left shape is straight and the right shape has the form described) — i.e., we can find a bijection between Zelevinsky's pictures and LR tableaux.

This is one generalization of the LR rule, but in fact it's not the only one found by Zelevinsky — next class we'll see another rule that gives a combinatorial rule for $s_{\lambda/\mu} \cdot s_\nu = \sum \bullet \cdot s_\gamma$ (where we multiply any skew shape by any straight shape — if μ is empty we get LR coefficients, while if ν is empty we should get LR coefficients with a different labelling).

§35 December 7, 2022

The Littlewood–Richardson rule is a rule for the coefficients $c^\lambda_{\mu\nu}$. There are two ways to interpret these coefficients — either as $\langle s_\lambda, s_\mu s_\nu \rangle$ (the coefficients from expanding $s_\mu s_\nu$) or as $\langle s_{\lambda/\mu}, s_\nu \rangle$.

Theorem 35.1

$c^\lambda_{\mu\nu}$ is the number of Littlewood–Richardson tableaux of shape λ/μ and weight ν.

Definition 35.2. A SSYT T is a LR-tableau if its Hebrew reading word is a Yamanouchi word.

Example 35.3

The tableaux

$$T = \begin{array}{ccccccc} & & & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} \\ & \boxed{1} & \boxed{2} & \boxed{2} & \boxed{2} & & \\ \boxed{1} & \boxed{2} & \boxed{3} & \boxed{3} & & & \end{array}$$

has Herew reading word 111122213321.

First, why are we reading right to left? Other ancient civilizations used different ways to read and write; for example, the Chinese empire traditionally read by columns, so we could also talk about the Chinese reading word where we read by columns, top to bottom and right to left. So we could also define the Chinese reading word of a tableaux.

Example 35.4

Our tableaux would have Chinese reading word 111212323121. This is also a Yamanouchi word.

This is not a coincidence!

Exercise 35.5. Check that we can replace the Hebrew reading with the Chinese reading and we would get the same rule — T is a Littlewood–Richardson tableaux if and only if it is a SSYT and its Chinese reading word is Yamanouchi.

Remark 35.6. There will be an optional problem set 3 to solve any of the exercises mentioned in class and submit it by the last day of class. This will not affect your letter grade, but it may affect the quantifier after the letter grade (you can make a difference between an A and A+).

(Another exercise is to know these exercises — anything mentioned in class but not proved is a valid problem.)

This is equivalent, but the Chinese version is a bit easier to check. This is because the entries of a SSYT are *strictly* increasing in columns. So if we read in teh Chinese way and the Yamanouchi condition fails at some point, then as we keep reading it will fail even worse. This means it's enough to check the condition not for every initial subword, but only at the moments when we reach the bottom of a column. This gives us an easier version to check:

Theorem 35.7

$c^\lambda_{\mu\nu}$ is the number of $T \in \mathrm{SSYT}(\lambda/\mu, \nu)$ such that if we cut the SSYT by any vertical line and call the part of the tableaux to the right of the line $T_{\geq j}$, then $\mathrm{wt}(T_{\geq j})$ satisfies the Yamanouchi condition (the number of 1's is greater than the number of 2's, and so on) — i.e. this weight is a partition.

Another benefit of this rule is that it's easier to generalize and easier to prove; our goal for today is to generalize it and then prove it.

Recall that last lecture, we mentioned Zelevinsky's picture rule for a more general set of coefficients $c(\lambda, \mu, \gamma, \nu) = \langle s_{\lambda/\mu}, s_{\gamma/\nu}\rangle$. Recall that by the property of Hall's inner product, this equals

$$\langle s_\nu \cdot s_{\lambda/\mu}, s_\gamma\rangle.$$

So these same coefficients occur if we want to multiply any skew Schur function by a straight shape and expand it in the basis of straight shapes —

$$s_{\lambda/\mu} s_\nu = \sum_\gamma c(\lambda, \mu, \gamma, \nu) s_\gamma.$$

(We can see this generalizes *both* ways to think of LR coefficients.)

Exercise 35.8. Show that in these two special cases $\mu = 0$ and $\nu = 0$, the picture rule is equivalent to the classical rule.

Theorem 35.9 (Zelevinsky)

$c(\lambda, \mu, \gamma, \nu)$ is the number of SSYT of shape λ/μ such that $\mathrm{wt}(T)+\nu = \gamma$, such that for any j, $\mathrm{wt}(T_{\geq j})+\nu$ is a partition.

So we take the Chinese formulation and modify the weight by adding ν. If ν is empty, it's clear we get the original version back.

Student Question. What does it mean to add partitions?

Term-wise, as vectors (add their first part, and so on).

Our goal for today is to prove this rule. This proof is by Steinbridge and Berenstein–Zelevinsky.

Before we prove the rule, we'll first formulate the theorem a bit more carefully. We are going to fix n, and suppose ν has at most n parts; so $\nu = (\nu_1, \dots, \nu_n)$, where some entries can be 0. We then let ρ be the staircase partition $\rho = (n-1, n-2, \dots, 1, 0)$, and we let λ/μ be *any* skew shape (it can have an arbitrary number of parts). We will fix the number of variables and try to find a rule for $s_{\lambda/\mu}(x_1, \dots, x_n) \cdot s_\nu(x_1, \dots, x_n)$ expanded in the basis of regular Schur functions in n variables $s_\gamma(x_1, \dots, x_n)$. (We can get to the infinite-variable case by taking the limit, but the proof requires us to have a finite number of variables.)

There are two definitions of Schur functions — a combinatorial and classical definition. We will use both — we will assume that s_ν and s_γ are defined by the classical definition. But the classical definition doesn't work for skew shapes, so $s_{\lambda/\mu}$ will be combinatorially defined, as a sum over SSYT.

Recall that by definition

$$s_\nu = \frac{a_{\nu+\rho}}{a_\rho},$$

where

$$a_{\alpha_1,\dots,\alpha_n} = \det(x_i^{\alpha_j})_{i,j\in[n]}.$$

(In particular this is anti-symmetric with respect to permuting either x or α, and a_ρ is the usual Vandermonde determinant.) We are going to use the same definition for s_γ; now we can clear the denominators. So we are going to prove the following:

Theorem 35.10

We have

$$s_{\lambda/\mu}(x_1, \dots, x_n) \cdot a_{\mu+\rho} = \sum a_{\nu+\mathrm{wt}(T)+\rho}$$

where $T \in \mathrm{SSYT}(\lambda/\mu, n)$ (where n denotes that all entries are at most n, or in other words that the weight has at most n parts — entries are in $[n]$ — this is different than the weight notation) such that for any j, $\mathrm{wt}(T_{\geq j}) + \nu$ is a partition.

We will call this condition CY (for 'Chinese Yamanouchi'). In this formula, these things are all n-vectors, and we add them componentwise.

Proof. We can expand the determinant as a sum over permutations and SSYT as

$$\mathrm{LHS} = \sum_{w\in S_n} \sum_{T\in\mathrm{SSYT}(\lambda/\mu,n)} (-1)^{\ell(w)} x^{\mathrm{wt}(T)+w(\nu+\rho)}$$

(where we combined the combinatorial definition and the determinant). We can rewrite this as

$$\sum_w \sum_T (-1)^{\ell(w)} x^{w(\mathrm{wt}(T)+\nu+\rho)}.$$

The difference is that here we're applying w to $\nu+\rho$ and also $\mathrm{wt}(T)$, instead of just $\nu+\rho$. We can do this because the Kostke numbers don't chnage under permutation fo the weight (we proved this earlier using Bender–Knuth involutions, which will be a crucial part of this proof). But now in this version, we can switch the order of summation — we can rewrite this as

$$\sum_{T\in\mathrm{SSYT}(\lambda/\mu,n)} \sum_{w\in S_n} (-1)^{\ell(w)} x^{\cdots}.$$

But this is just

$$\sum_T a_{\mathrm{wt}(T)+\nu+\rho}.$$

Now we can compare this with the right-hand side of the theorem — there we also had an alternating sum of determinants, but with the additional condition. So somehow this sum includes all terms here, but also some additional terms where this condition fails.

So the idea is to construct a sign-reversing involution on the set that cancels all tableaux where our condition fails, so that we can cancel all terms in this formula. You may wonder how we can have a sign-reversing involution when there are no signs; but the signs are coming from the fact that $\mathrm{wt}(T)+\nu+\rho$ may not be a partition, so its parts may not be decreasing. And a is anti-symmetric with respect to permutation of parts, so if we arrange them in a non-increasing order, we might actually get signs if we express them in terms of the a_α with decreasing entries.

So now let's try to construct a sign-reversing involution on SSYT of a given shape λ/μ (and entries at most n) such that the condition (CY) fails.

What this means is we have a tableaux and we cut it at its jth column and consider $T_{\geq j}$, and there exists k such that the number of k's in $T_{\geq j}$ plus ν_k is less than the number of $(k+1)$'s in $T_{\geq j}$ plus ν_{k+1} — we can find at least one pair (k,j) such that this holds.

There might be several pairs (k,j) with this condition. So among all these places, we want to first maximize j, and after that we want to minimize k. We want to find a pair (j,k) with j as large as possible, and then pick k as small as possible (among the pairs with equal j).

This means we have this inequality, but if we replace j by $j+1$ then the same is not true — then the number of k's in $T_{\geq j+1}+\nu_k$ is at least the number of $(k+1)$'s in $T_{\geq j+1}+\nu_{k+1}$.

So we are comparing T_j and something which is slightly larger, T_{j+1}. So the only difference is in this column, where entries are all strictly increasing — in this column we have at most one k, and at most one $k+1$. This means the left-hand side can increase only by 1, and the right-hand side can also only increase by at most 1. So the difference can be at most 1 (they are either equal or differ by 1); but the difference must be negative and nonnegative. So this implies we must have equality for $j+1$, and an inequality with -1 for j.

In other words, since the jth column of T contains at most one k, and at most one $k+1$, we deduce that

$$\#k \text{ in } T_{\geq j}+\mu_k = \#(k+1) \text{ in } T_{\geq j}+\nu_{k+1}-1,$$

and that the only possibbility is that the jth column doesn't contain any k's, and it contains exactly one $k+1$. These two observations will be useful.

We are now ready to define the sign-reversing involution. We first take our sub-tableaux on one side $T_{\geq j}$, and the other side $T_{<j}$ to the left formed by the first $j-1$ columns of T.

The next thing we will do is apply a Bender–Knuth involution, which we constructed at the beginning of the class using toggles and some operations transforming k to $k+1$.

Definition 35.11. The **Bender–Knuth operation** σ_k is a map $S \mapsto \tilde{S}$ obtained by changing some k's to $(k+1)$'s and vice versa, such that S and $\tilde{S}$ are SSYT of the same shape, and if $\mathrm{wt}(S) = (\beta_1, \ldots, \beta_n)$, then the $\mathrm{wt}(\tilde{S})$ is obtained by switching β_k and β_{k+1}, which we can write as $s_k(\beta)$.

We are now going to apply σ_k not to teh whole tableaux, but to $T_{<j}$ — so we apply Bender–Knuth *only* to the sub-tableau.

Let $T \to T^*$ be the map where T^* is obtained from T by replacing $T_{<j}$ with $\sigma_k(T_{<j})$ — in other words, we take the left sub-tableaux and replace some k's and $(k+1)$'s with each other.

Some observations:

- T^* is a SSYT. This is not totally obvious — we replaced some k's by $(k+1)$'s and some $(k+1)$'s by k's. If we had a SSYT the two parts don't change, but some inequalities along the cut might actually fail — because we could replace a k with a $(k+1)$, and if we had a k to its right, then we would have a failure. But we don't have any k's in this jth column, so there are no places we could fail — even if we increase some k's next to the cut to $k+1$, there is no k in the column so we still satisfy the SSYT condition.
- $T \mapsto T^*$ is an involution — it does not affect the $T_{\geq j}$ part, so if we apply the operation to T^*, we will find the same pair (j, k), and the BK operation is an involution.
- We need to check that the weight of this tableaux is what we need. But we can see that $\mathrm{wt}(T_{<j}) = s_k(\mathrm{wt}(T^*_{<j}))$ and $\mathrm{wt}(T_{\geq j}) + \nu = \mathrm{wt}(T^*_{\geq j}) + \nu$, because this part does not change at all (none of the entries here changed, so we have the same weight).

  On the other hand, this equals $s_k(\mathrm{wt}(T^*_{\geq j}) + \nu)+$ — in this vector, the number of k's is almost the same as the number of $k+1$'s, so if we switch the k and $(k+1)$st entry we get almost the same thing. But they differ by 1 — so swtiching them means we change our expression by $(0, 0, \ldots, 1, -1, 0, \ldots)$ where the 1 is in the kth position and -1 in the $(k+1)$st.

  This implies that
  $$\mathrm{wt}(T) + \nu + \rho = s_k(\mathrm{wt}(T^*) + \nu + \rho)$$
  (the additional ρ gives you this vector).

  This is exactly what we wanted to get — because the weight of this expression differs by one simple transposition, and so $a_{\mathrm{wt}(T)+\nu+\rho} = -a_{\mathrm{wt}(T^*)+\nu+\rho}$. So they are going to exactly cancel each other in the sum of determinants (because a is alternating with respect to permutations of the vector).

  So we have a sign-reversing involution that cancels all semistandard Young tableau where CY fails, and after all cancellations we get exactly the LR rule.

$\square$

Exercise 35.12. There will be several exercises to check that various specializations of this rule give you the classicla rule — you can specialize μ to 0, or you could specialize ν to 0, and you should check the correspondences betwen the classical formulation adn the generalized version.

§36 December 9, 2022

Previously, we saw that

$$c^{\lambda}_{\mu\nu} = \#\{\text{LR-tableaux of shape } \lambda/\mu \text{ and weight } \nu\}.$$

A tableau T is a LR-tableau if its Semitic reading word (obtained by reading from right to left, top to bottom) is a Yamanouchi word.

In particular, this implies the first row can only have 1's; the second row can have some number of 2's, and then some number of 1's; the third can have some number of 3's, 2's, and 1's; and so on. So if we say there are a 1's, b 2's, c 1's, d 3's, e 2's, f 1's, and so on, then we need to have $a \geq b$, $b \geq d$, $a + c \geq b + e$, and so on. So we can write our conditions as some linear inequalities.

This is a nice rule, but it doesn't explain many symmetries of LR–coefficients.

§36.1 Berenstein–Zelevinsky Polytopes

Berenstein–Zelevinsky addressed two questions. First, they wanted to reformulate the rule in terms of polytopes (as the number of lattice points in a certain convex polytope — similarly to how the Gelfand–Tsetlin polytopes' lattice points count the Kostke numbers). And we would like to do this in a way that explains some symmetries of the coefficients.

In order to do this, we first fix n. We will assume λ, μ, and ν all have exactly n parts (some of which can be 0), so we will think of them as vectors in $\mathbb{Z}^n$.

We first want to construct the Gelfand–Tsetlin parterns — we can think of a tableaux T of shape λ/μ as a series $\mu \subseteq \mu^{(1)} \subseteq \mu^{(2)} \subseteq \cdots \subseteq \mu^{(n)} = \lambda$, where each is obtained by adding a horizontal strip to the previous (corresponding to the boxes with entries i).

We then write out

$$\mu = \mu_1, \mu_2, \mu_3, \ldots, \mu_n.$$

Then we write out

$$\mu^{(1)} = \mu_1^{(1)}, \mu_1^{(2)}, \ldots$$

so that these numbers are interlaced (i.e. $\mu_1^{(1)} \geq \mu_1 \geq \mu_2^{(1)}$). We keep on doing this; each row has n elements, and the rows move to the left. (For regular GT patterns we had a triangular shape; here we instead have a rhombus, because we're working with skew shapes.) So we have a rhombus filled with numbers, which decrease along two diagonals.

We can easily figure out entries in exactly half of this rhombus — in the first row we can have only 1's. The first row is given by the first diagonal; so then all of $\mu_1^{(1)} = \mu_2^{(2)} = \cdots = \lambda_1$. Similarly $\lambda_2 = \cdots = \mu_2^{(2)}$, and so on. So everything happening inside the bottom triangle is trivial; the only place interesting things happen is in the upper one.

As usual, ν_i is also the $(i+1)$st row sum minus the ith row sum.

These are all the constraints coming from the SSYT, but here we have additional inequalities. It's actually easy to figure out these numbers — a is the number of 1's in the first row, which is $\mu_1^{(1)} - \mu_1$. Then b is the number of 2's in the second row, which is $\mu_2^{(2)} - \mu_1^{(2)}$. And similarly $c = \mu_2^{(1)} - \mu_2$, and we can find d, e, f, and so on similarly.

Example 36.1

The condition $a \geq b$ means that the difference between a certain two entries is at least the difference between another two entries.

So now all the Yamanouchi conditions are equivalent to some linear inequalities on the entries of this pattern.

So this means we can define a polytope given by all these linear conditions.

Definition 36.2. The Berenstein–Zelevinsky polytope $\mathrm{BZ}_{\mu\nu}^{\lambda}$ is the set $\{(p_{ij}) \subseteq \mathbb{R}^{\binom{n+1}{2}} \mid \text{all linear inequalities}\}$ of (not necessarily integer) points satisfying all the linear inequalities (the interlacing conditions, the Yamanouchi conditions, and the equalities given by μ and ν).

Remark 36.3. There is a more general construction of Berenstein–Zelevinsky polytopes for any Lie algebra.

The Littlewood–Richardson rule now says that the LR–coefficients are the number of integer lattice points in $\mathrm{BZ}_{\mu\nu}^{\lambda}$.

Remark 36.4. We should be careful because there are examples of triples of integer weights where BZ is not a lattice polytope — it might have non-integer vertices. This means usual Erhart theory does not directly apply.

Exercise 36.5. Find an example where this polytope is not an integer polytope.

We would like to rewrite this polytope in a change of variables that makes these symmetries more explicit.

To talk about these symmetries, it's convenient to slightly extend the class of objects we're looking at — in particular we will allow *negative* entries in λ, μ, and ν (as in the GL_n point of view).

Definition 36.6. A **weight** is $\lambda = (\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n)$ for $\lambda_i \in \mathbb{Z}$. (In representation theory, they're called *integer dominant weights.*)

So in other words, we want to extend $s_\lambda(x_1, \dots, x_n)$ to *weights*. We can do this easily using the property that

$$s_{\lambda + a(1,1,\dots,1)}(x_1, \dots, x_n) = (x_1 \cdots x_n)^a s_\lambda(x_1, \dots, x_n),$$

where a can be any integer. Now our polynomials are Laurent polynomials instead (where we allow negative powers of x), and they form a linear basis there as well.

We still define the coefficients in this more general setting as

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

We also define $\lambda^* = (-\lambda_n, -\lambda_{n-1}, \dots, -\lambda_1)$, and we define

$$c_{\lambda\mu\nu} = c_{\mu\nu}^{\lambda^*}.$$

Fact 36.7 — $c_{\lambda\mu\nu}$ is symmetric in all permutations of λ, μ, and ν.

Proof. This is the coefficient of $s_{(0,0,\dots,0)}$ (i.e. the constant term) in the Schur expansion of $s_\lambda s_\mu s_\nu$. $\square$

We will rewrite our polytope in different coordinates so that some of these symmetries become trivial.

§36.2 Berenstein–Zelevinsky Triangles

Example 36.8

When $n = 5$, we draw a triangle with 8 dots on each side. Then we draw a line through every second dot (so we have three lines). There are also dots in the middle, at each intersection.

In general, there are $2(n-1)$ dots on each side. These dots are not just dots, but numbers — so we have a variable a_d for each dot d. These dots should satisfy some linear conditions:

- $a_d \geq 0$ for all d. (When talking about a polytope we allow arbitrary real numbers; when talking about lattice points they are integers.)
- Certain boundary conditions — the triangle has three sides, which we call the λ side (on the left), μ side (on the right), and ν side (on the bottom, with all sides directed counterclockwise. On the λ side if we set the variables $x_1, \ldots, x_{2(n-1)}$, then we should have $x_1 + x_2 = \lambda_1 - \lambda_2$, $x_3 + x_4 = \lambda_2 - \lambda_3$, and so on; in general $x_{2i-1} + x_{2i} = \lambda_i - \lambda_{i+1}$ for all $1 \leq i \leq n-1$.

  We have similar conditions on the μ side and the ν side — if we call the μ side $y_1, \ldots, y_{2(n-1)}$ with $y_1 = x_{2(n-1)}$, then we should have $y_{2i-1} + y_{2i} = \mu_i - \mu_{i+1}$, and if we call the ν side $z_1, \ldots, z_{2(n-1)}$ then we should have $z_{2i-1} + z_{2i} = \nu_i - \nu_{i+1}$. Here we assume that $|\lambda| + |\mu| + |\nu| = 0$ (note that these are the sums $\lambda_1 + \cdots + \lambda_n$, which may be negative).
- The *hexagon condition* — for every little hexagon, let the 6 numbers sitting at the vertices of the hexagon be t, u, v, w, p, q in order. These satisfy:
  - $t + u = w + p$.
  - $u + v = p + q$.
  - $u + w = q + t$.

  In other words, every side has the same sum as the opposite edge.

Definition 36.9. The polytope $\tilde{\mathrm{BZ}}(\lambda, \mu, \nu)$ is the polytope $\{(a_{ij}) \subseteq \mathbb{R}^N\}$ such that all these linear conditions are satisfied. (Here N is the total number of dots, which can be calculated explicitly.)

Theorem 36.10 (Berenstein–Zelevinsky)

The number of lattice points in this polytope equals $c_{\lambda\mu\nu}$.

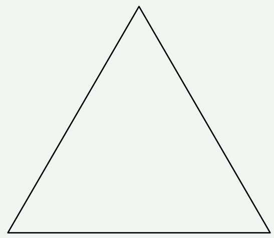
Proof. This polytope is linearly isomorphic to the polytope defined at the beginning of the lecture $\mathrm{BZ}_{\mu\nu}^{\lambda^*}$, where this linear isomorphism preserves all lattice points.

Exercise 36.11. Figure out how this linear isomorphism works explicitly — i.e., express the a_d in terms of the p_{ij}.

□

Example 36.12

Suppose $n = 4$, so our shape is:



(We have 6 dots on each edge.) Suppose read from top to bottom, our vertices have values (0), $(1, 0)$, $(1, 2)$, $(2, 1, 0, 3)$, $(1, 2, 1)$, $(2, 2, 1, 2, 1, 3)$.

Then we know $\lambda_1 - \lambda_2 = 2 + 1$, $\lambda_2 - \lambda_3 = 2 + 1$, and so on; this defines λ, μ, and ν up to adding $a(1, 1, \ldots)$. We also need to make sure we pick λ, μ, and ν such that the sum of all parts is 0. This can be done by setting $\lambda = (-7, -10, -13, -15)$, $\mu = (10, 9, 4, 0)$, and $\nu = (11, 7, 4, 0)$.

Here μ and ν are actual partitions; λ is not, but $\lambda^* = (15, 13, 10, 7)$ is. So this should be one of the Gelfand–Tsetlin patterns that contributes to $c_{\lambda\mu\nu}$ or equivalently $c_{\mu\nu}^{\lambda^*}$.

§36.3 Some Symmetries

We can immediately see that we can rotate the triangle, so we have cyclic symmetry — $c_{\lambda\mu\nu} = c_{\mu\nu\lambda} = c_{\nu\lambda\mu}$.

We can also reflect the triangle. Then all the conditions are preserved and we get μ, ν, and λ, but directed in the opposite way; so then we get that this equals $c_{\mu^*\lambda^*\nu^*}$ and its cyclic rotations $c_{\lambda^*\nu^*\mu^*}$ and so on. But no obvious transformation of this triangle will result in $c_{\lambda\mu\nu} = c_{\lambda\nu\mu}$. We know this equality has to hold because $s_\mu s_\nu = s_\nu s_\mu$, but somehow there is no linear transformation of the Berenstein–Zelevinsky triangles that result in this.

So somehow this commutativity is the only non-obvious symmetry.

§36.4 Knutson–Tao's Honeycombs

There is a nice way to draw these Berenstein–Zelevinsky patterns in the plane, found by Knutson and Tao.

A honeycomb is a picture drawn on the plane, with certain lines. We only allow three directions for these lines — lines can be vertical, $30°$ to the horizon, or $-30°$. These hexagons do *not* have to be regular; we require that the direction of the lines are the same as in a regular hexagon, but the line lengths can be arbitrary; and we can even allow degenerate cases where some of the lines collapse into a single point. (So a hexagon can become a pentagon or a quadrilateral or even a single point.)

We also have boundary rays going in the three directions (30, 150, and 270). The coordinates of these boundary rays are given by $\lambda_1, \ldots, \lambda_n, \mu_1, \ldots, \mu_n, \nu_1, \ldots, \nu_n$ clockwise (with λ on the left).

We think of this picture as the set of points $(x, y, z) \in \mathbb{R}^3$ with $x + y + z = 0$. These three directions are given by requiring things in each of these three coordinates — so the line corresponding to λ_1 is the line $(\lambda_1, *, *)$; similarly for ν it's $(*, \nu_i, *)$, and for ν it's $(*, *, \nu_i)$.

Definition 36.13. The honeycomb is *integer* if the coordinates of all the boundary rays, and the lines in the middle are all integers.

Theorem 36.14 (Knutson–Tao)

$c_{\lambda\mu\nu}$ is the number of integer honeycombs for λ, μ, and ν.

To convert a BZ pattern into a honeycomb, we can take the lengths of segments in the honeycomb; we then put these letters in the BZ pattern (the conditions are equivalent — for example, the $x_1 + x_2$ condition tells us the distance between two rays is $\lambda_1 - \lambda_2$, which is what we want here).

§37 December 12, 2022

§37.1 Honeycombs

We draw honeycombs on the plane $\{x + y + z = 0\} \subseteq \mathbb{R}^3$. We use lines of three directions, given by fixing the x, y, or z coordinate (lines in the 150° direction fix x, 30° fix y, 270° fix z).

A **honeycomb** is a picture drawn on the plane with line segments of these three types, where there are three types of infinite rays — on the left side we have infinite rays with x-coordinate λ_1, λ_2, ... (from bottom-left to top-right), on the right we have $\mu_1, \ldots, \mu_n$ (top to bottom), and $\nu_1, \ldots, \nu_n$ on the bottom (right to left). The honeycomb is integer if all line coordinates are integers.

It's easy to see a bijection between honeycombs and everything else.

A BZ triangle is an array of numbers of the shape described. These numbers can be read off the honeycombs — they are the lengths of all line segments in a honeycomb. (The little length of the line at the top goes to the top dot, then b and c below it to the two dots below.)

You can write LR tableau as a GT pattern, with a rhombus kind of shape filled with numbers. (We then have λ^* on the bottom.) The Yamanouchi condition guarantees all the lines on the bottom are λ_i^*, so everything interesting is in a triangle with a GT pattern plus Yamanouchi conditions.

We look at lines in the / direction (the μ direction). If we record the coordinates of these lines, the first should be $\mu_1, \ldots$; the next should be interlaced with the μ_i; the next are interlaced, and so on. IF you know λ, μ, and ν and the coordinates of lines in one direction, then you can figure out everything else; and you get these from the entries of the thing. We also have the extra condition that all line segments should have nonnegative lengths, and this exactly matches the Yamanouchi conditions coming from Yamanouchi words.

§37.2 Puzzles

To think about puzzles we want to think about the Grassmanian interpretation, where $\lambda, \mu, \nu \subseteq k \times (n-k)$. We want to define $\tilde{c}_{\lambda\mu\nu} = c_{\mu\nu}^{\lambda^\wedge}$ where $\lambda^\vee$ is the complement of λ in the rectangle, rotated. It's the coefficient of $s_{k\times(n-k)}$ in $s_\lambda s_\mu s_\nu$.

A **puzzle** is made of three puzzle pieces. There are three types — triangles and rhombuses — with numbers on the sides (a 000 triangle, a 111 triangle, and a 1010 rhombus). You can rotate but you cannot reflect — note that the mirror image of the rombus is not allowed (we have 1 on the top and bottom, with the lilnes directed 60°).

Given a partition in $k \times (n-k)$, we can convert it to a 01-vector with k and $n-k$ 0's and 1's — think of it as a lattice path, every time you go left you write 0, every time you go down you write 1. This gives some sequence of 0's and 1's. So nwo we have three vectors of length n with exactly k 1's.

Our puzzle will be a triangle tiled by pieces of these three types. If $n = 4$ and $k = 2$, then:

We have three sides (an upwards-facing equilateral triangle). The sides have length 4, so 5 dots. The segments are labelled with the elements of λ, μ, ν (as 01-vectors), with λ from 210 → 90, μ from 90 → 330, ν 330 → 210. The rule is that we can only glue two pieces if they have the same number on the shared side.

(For example, $\lambda = 0, 1, 1, 0)$, $\mu = 0, 1, 1, 0$, $\nu = 0, 0, 1, 1$, and a 0-triangle at the top, 1-triangle and rhombus in the next row, triangle triangle triangle rhombus in the third, rhombus rhombus triangle triangle triangle in the last. (We color the 1-triangles green, so we have a block of four of these on the left.)

Claim 37.1 — These are in bijection with all the other stuff.

It's a bit hard to see a honeycomb in this picture, because it is a very degenerate case. But we can draw another puzzle in the case $k = 1$ where n is very large, so we only have one 1 on each side. Then there will be only on epuzzle, and it will look like a triangle with some rhombi leading into it from those 1's, and 0's everywhere else. SO when $k = 1$ there is exactly one green triangle, three rays of rhombuses going to each of the three sides of the large triangle, and the big areas have every tile being a 0 triangle. We can see this picture looks like a honeycomb with three rays λ_1, μ_1, and ν_1, which are the positions of the 1's on the sides. The pictures are basically the same except that instead of lines we have ribbonized rhombi. So we can think of a puzzle as a ribbonized honeycomb.

More generically we can start with a honeycomb, and then thicken all its lines — we take our honeycomb and replace every section with a chain of rhombuses, and we have a little green triangle where these segments meet. Then we place this inside a large triangle, with 1's at the corresponding positions and 0 everywhere else, and our ribbons being chains of rhombuses. The number of rhombuses is the length of the line segment — if the segment has length a, we put a rhombuses. You can check that everything works, and this gives a correspondence.

(The number of green triangles is exactly the number of junctions in the picture.)

But honeycombs are nice because you can see a symmetry where you replace 0's by 1's and take the mirror image of the picture (because of the rhombus orientation) — so there will be a dual honeycomb coming from the same sort of puzzle. So if we take the conjugate partitions, we get a dual puzzle corresponding to a dual honeycomb, although the honeycomb and dual honeycomb look very different if you don't think about puzzles. (We have to eribbonize every lien segment in the honeycomb to convert it to puzzles, then switch 0 and 1, and then un-ribbonize it.)

§38 December 14, 2022

We've seen several ways to think about the Littlewood–Richardson coefficients $c^{\lambda}_{\mu\nu}$. One way, from the representations of GL_n, is to fix n and think of λ, μ, and ν as integer vectors with weakly decreasing entries (which may be negative). From this point of view it's natural to think about Berenstein–Zelevinsky triangles and Knutson–Tao honoeycombs, where the Littewood–Richardson coefficient counts the number of integer honeycombs with given boundary rays.

Another point of view comes from the geometry of the Grassmanian. Here we fix $0 \le k \le n$ and think of (λ, μ, ν) as a triple of partitions which fit inside $k \times (n-k)$. Then we can associate these partitions with $\{0, 1\}$-vectors with k 1's (thinking of our partition as a lattice path), or equivalently as k-element subsets of $[n]$ (adding an extra 1 to each entry of the partition). From this point of view it's natural to think about puzzles (with 0's and 1's on the sides, corresponding to these subsets).

Question 38.1. What can we say about the triples (λ, μ, ν) such that $c^{\nu}_{\lambda\mu} \neq 0$?

In the honeycomb point of view, we fix the positions of the boundary rays; we then want to see whether we can find an integer honeycomb with given positions of boundary rays.

§38.1 Horn's Problem

Interestingly, this is related to a problem which looks like it's from a very different area of mathematics. This problem is not even about Young diagrams or partitions or standard Young tableau; it is about matrices.

Suppose we have three $n \times n$ Hermitian matrices with $A + B = C$. A complex matrix is *Hermitian* if $A^* = A$ — A^* is the complex conjugate of $A^\intercal$. A fact about Hermitian matrices is that their eigenvalues are all real; suppose the eigenvalues of A are $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

Question 38.2 (Horn's Problem)**.** Describe all possible triples $(\lambda, \mu, \nu) \in \mathbb{R}^{3n}$ such that there exist Hermitian matrices with $A + B = C$ with given eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$, $\mu_1 \geq \cdots \geq \mu_n$, and $\nu_1 \geq \cdots \geq \nu_n$ respectively.

First, let's try to find some conditions on these triples of sequences. One observation is that if $A + B = C$, then $\text{tr}(A) + \text{tr}(B) = \text{tr}(C)$. The trace is the sum of eigenvalues, so we immediately get

$$\sum \lambda_i + \sum \mu_i = \sum \nu_i.$$

Another fact is that if we add two Hermitian matrices, the largest eigenvalue of the sum is at most the sum of the largest eigenvalues; this means we must have $\lambda_1 + \mu_1 \geq \nu_1$.

There is an even more general version of this:

Fact 38.3 (Weyl's inequality) — We have $\lambda_{1+i} + \mu_{1+j} \geq \nu_{1+i+j}$ for all i and j such that $i, j \geq 0$ and $i + j < n$.

Note that all conditions we've written so far are *linear* conditions. In fact, this is true in general:

Theorem 38.4

There exists a certain polyhedral cone $\text{Klychko}(n) \subset \mathbb{R}^{3n}$ such that (λ, μ, ν) is a solution to Horn's problem if and only if it belongs to $\text{Klychko}(n)$.

In other words, the set of solutions forms a certain polyhedral cone (all conditions are linear inequalities).

In fact, there is a reason we used the same Greek letters here as in our partitions:

Theorem 38.5

Suppose that $\lambda, \mu, \nu \in \mathbb{Z}^n$ (with weakly decreasing parts). Then $c^{\lambda}_{\mu\nu}$ is nonzero if and only if $(\lambda, \mu, \nu) \in \text{Klychko}(n) \cap \mathbb{Z}^{3n}$.

In other words, the set of solutions to our problem is exactly the set of integer solutions to Horn's problem.

The problem started with Horn. Klychko almost proved this theorem, modulo a saturation hypothesis. The final step was done by Knutson and Tao, who solved the saturation hypothesis — this was the reason they invented honeycombs. More specifically, they proved the following result:

Theorem 38.6 (Klychko's saturation conjecture)

Suppose that $(\lambda, \mu, \nu) \in \mathbb{Z}^n$, and $k \in \mathbb{Z}_{>0}$. Then if $c_{k\lambda, k\mu, k\nu}$ is nonzero, then $c_{\lambda,\mu,\nu}$ is nonzero.

So in other words, if we can scale our three partitions and get a nonzero coefficient, then the original coefficient was also nonzero.

We can formulate this theorem in terms of honeycombs as well — suppose we fix the boundary rays λ, μ, and ν. Then the claim is that if we can scale the picture and find an integer honeycomb, then we can shrink it back to get an integer honeycomb. In fact, they proved a stronger result — if given integer boundary rays we can find *some* honeycomb (with real edge lengths), then we can actually find an integer honeycomb with the same boundary rays.

Honeycombs are useful here for the following reason: suppose we have found some honeycomb, and we know all boundary rays are integers. We want to somehow deform this honeycomb without changing any of the boundary rays. And an important thing is to find a cycle. If we find a hexagon (in the $n = 3$ example), we see that this hexagon has one degree of freedom — we can inflate or dilate the hexagon without changing the boundary rays. We can keep doing this — as long as we have a degree of freedom (from some cycle), we keep doing this, until some edge collapses to a point.

In this example, we now have a pentagon. And the claim is that if we have the most degenerate honeycomb, then all coordinates can be expressed as linear combinations of the boundary rays with integer coefficients. This is because we know that the sum of the coordinates of three lines meeting at a point is 0. In the most degenerate case the graph is a tree (whenever we collapse an edge, the vertex sort of disappears, and then we have a tree); and we cn then reconstruct the coordinates of all edges from the boundary, and everything has to be an integer.

§38.2 The Klychko Cone

Now we know the problem of finding nonzero Littlewood–Richardson coefficients is the same as describing the Klychko cone.

For Hermitian matrices, the classification was proved by Klychko; the connection to coefficients is as above.

Theorem 38.7

For $(\lambda, \mu, \nu) \in \mathbb{R}^{3n}$, we have $(\lambda, \mu, \nu) \in \text{Klychko}(n)$ if and only if:

- $\sum \lambda_i + \sum \mu_i = \sum \nu_i$;
- $\lambda_1 \geq \lambda_2 \geq \cdots$, $\mu_1 \geq \mu_2 \geq \cdots$, and $\nu_1 \geq \nu_2 \geq \cdots$;
- $\sum_{i \in I} \lambda_i + \sum_{j \in J} \mu_j \geq \sum_{\ell \in L} \nu_\ell$, for all triples of subsets $I, J, K \in \binom{[n]}{k}$ (where $1 \leq k \leq n-1$) such that $c_{IJ}^L \neq 0$, where $c_{IJ}^L = c_{\tilde{\lambda}\tilde{\mu}}^{\tilde{\nu}}$ is nonzero (these are the partitions corresponding to the subsets as above).

In particular $\lambda_1 + \mu_1 \geq \nu_1$ is a special case — the 1-element subset 1 corresponds to the empty partition, and $c_{\emptyset\emptyset}^{\emptyset} \neq 0$.

This may be somewhat confusing — we started with a problem on how to describe all triples of partitions such that their Littlewood–Richardson coefficient is nonzero. And in this description, we also have a condition based on triples corresponding to subsets with nonzero coefficient. This looks like it solves the problem by requiring we already know how to solve the problem.

But here we have two views of the coefficients at the same time. In the first case, we fix n, but the sizes of the parts can be arbitrarily large. But in the condition we fix n (it's the same n) and k, and then we only look at partitions that fit into $k \times (n-k)$ — this means in order to describe *infinitely many* triples, we only need to check finitely many conditions — and all partitions here will have strictly less than n parts. So this actually gives a recursive description — because in order to describe this we'd have to use the same theorem again — and this recursion converges very quickly.

This is an explicit description, but something's not completely satisfactory, because there's still some kind of recursion. It would be nice to have a description that doesn't require us to check the same inequality, but this is not known.

The case $n = 1$ is very easy; the next case is $n = 2$.

Example 38.8

Describe the Klychko cone for $n = 2$ — here we are looking at honeycombs with λ_1, λ_2, μ_1, μ_2, $-\nu_2$, $-\nu_1$ (starting from 210° and going clockwise).

There are some obvious conditions:

$$\lambda_1 + \lambda_2 + \mu_1 + \mu_2 = \nu_1 + \nu_2$$

and $\lambda_1 \geq \lambda_2$, $\mu_1 \geq \mu_2$, and $\nu_1 \geq \nu_2$. Then the only possible k is 1, and all the conditions we get are

$$\lambda_1 + \mu_1 \geq \nu_1$$

(corresponding to the case $s_\emptyset s_\emptyset = s_\emptyset$),

$$\lambda_1 + \mu_2 \geq \nu_2$$

(corresponding to $s_\emptyset s_{(1)} = s_{(1)}$), and

$$\lambda_2 + \mu_1 \geq \nu_2$$

(again from $s_{(1)} s_\emptyset = s_{(1)}$).

These three conditions can be equivalently formulated in the following form: suppose that $\ell = \lambda_1 - \ell_2$, $m = \mu_1 - \mu_2$, and $r = \nu_1 - \nu_2$ (i.e., the difference between our rays, or the boundary numbers in BZ triangles).

Here BZ triangles will be of the form a, b, c such that $a + b = \ell$, $b + c = m$, and $a + c = r$.

We can find such nonnegative numbers if and only if ℓ, m, and r satisfy the triangle inequality ($\ell \leq m + r$, $m \leq \ell + r$, $r \leq \ell + m$).

The case $n = 3$ can also be done by hand. But the point is that all conditions come from some problem about multiplying Schur polynomials (only restricting to partitions of very small size).