502 Project

Trent Osland

May 5, 2023

Abstract

In this paper we provide an overview of Knutson-Tao puzzles. We provide definitions and examples of puzzles as well as some interesting facts about them. We provide Knutson and Tao's proof that the puzzle rule can be used to find the Littlewood-Richardson coefficients.

We use [1], [3], and [2].

Always list citations so the numbers appear in order

1 Introduction

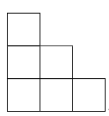
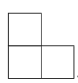
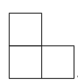
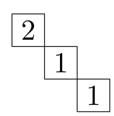
To begin, we state the main result about puzzles:

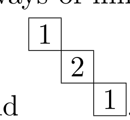
Theorem 1.1. *The Littlewood-Richardson coefficient* $c^{\lambda}_{\mu\nu}$ *is equal to the number of puzzles with boundary* $\Delta^{\lambda}_{\mu\nu}$.

1.1 Notation

We first recall the definition of the Littlewood-Richardson coefficients in terms of skew tableaux.

Definition 1.2. The Littlewood-Richardson coefficient $c^{\lambda}_{\mu\nu}$ is equal to the number of Littlewood-Richardson tableaux of shape λ/μ and content ν.

Example 1.3. If $\lambda =$ , $\mu =$ , $\nu =$ , then $c^{\lambda}_{\mu\nu}$ counts the ways of filling the skew tableaux λ/μ with content from ν, of which there are two ways: 

and 

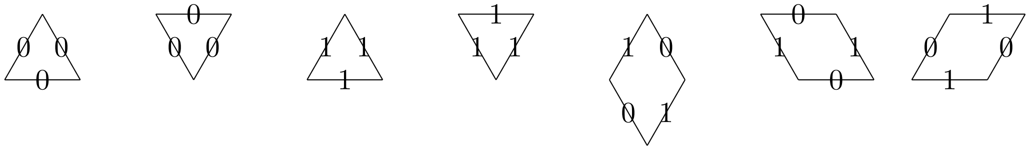
.

We now introduce the objects called "puzzles" which will give us a different way of computing the same $c^{\lambda}_{\mu\nu}$.

Definition 1.4. A **puzzle piece** is one of the following three planar figures with labeled edges:

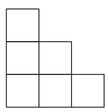
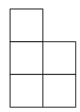
1. A unit triangle with all edges labeled 0.
2. A unit triangle with all edges labeled 1.
3. A unit rhombus (two unit triangles glued together), with the two edges clockwise of acute vertices labeled 0, and the other two edges labeled 1.

The 3 puzzle pieces are shown below in all possible orientations:



Definition 1.5. A **puzzle** is an equilateral triangle decomposed into puzzle pieces, examples of which can be seen on this page and the next.

Since we want to calculate the Littlewood-Richardson coefficients $c^{\lambda}_{\mu\nu}$ using puzzles, we need a correspondence between the partitions λ, μ, ν and puzzles. This is done by representing the partitions as binary strings in the following way: Starting at the top left corner of the Young diagram of λ, move along the outside with each step to the right represented by a 1 and each step down represented by a 0.

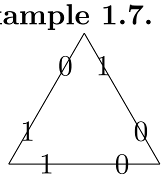
Example 1.6. If $\lambda_1 =$  then $\lambda_1 = 101010$. If $\lambda_2 =$  then $\lambda_2 = 10100$.

To write μ and ν as binary strings, use the same process, but start in the top left corner of the rectangle large enough to contain λ. This is because λ, μ, and ν need to all be binary strings of the same length. Thus in Example 1.3, $\lambda = 101010$, $\mu = 010101$, and $\nu = 010101$.

Now that we have our 3 partitions written as binary strings, these become the labels for the outer border of a puzzle. λ is the bottom side, μ is the top left side, and ν is the top right side. Once the outer border of a puzzle is set, there will then be some (non-negative integer) ways of completing the puzzle with the 3 puzzle pieces.

Some examples of puzzles are given below.

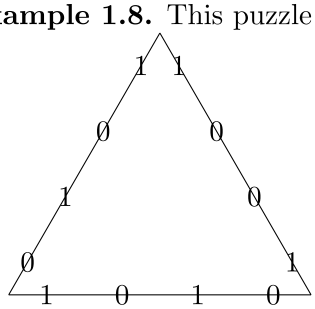
Example 1.7. The following puzzle has 0 ways of completing it:

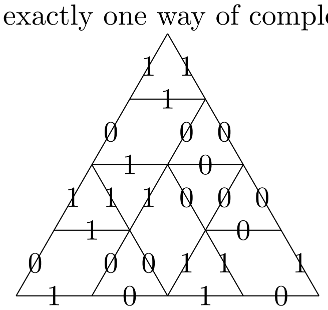


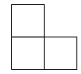
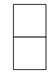
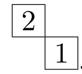
To see why this is true, observe that we have an issue at the very top, in that there is no puzzle piece with a 0 and 1 in that orientation.

This corresponds to $c^{\lambda}_{\mu\nu}$ where $\lambda = \mu = \nu = \square$. Thus $c^{\lambda}_{\mu\nu}$ is the number of ways of filling an empty skew tableaux with content 1. There are 0 ways of doing that, which matches the fact that there are 0 ways of completing the corresponding puzzle.

Example 1.8. This puzzle has exactly one way of completing it:

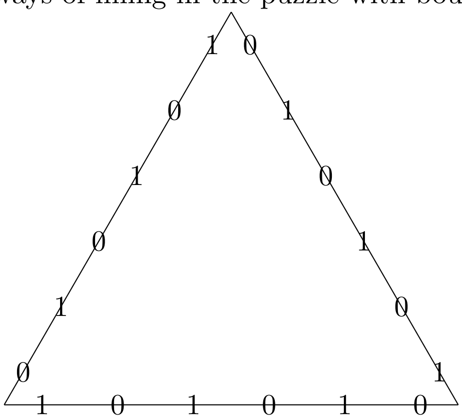




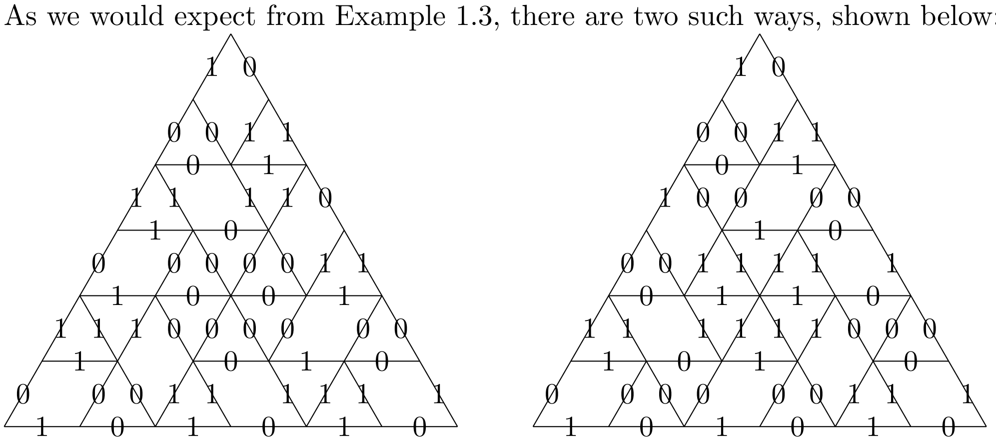
The equivalent characterization with skew tableaux is $\lambda =$ , $\mu =$ , $\nu =$ . Thus $c^{\lambda}_{\mu\nu}$ counts the number of ways of filling λ/μ with a 1 and a 2, of which there is 1 way: 

.

Example 1.9. For our final example, we return to Example 1.3 and look at the corresponding puzzle. In that case we have $\lambda = 101010, \mu = 010101, \nu = 010101$. Thus we need to find the number of ways of filling in the puzzle with boundary:



As we would expect from Example 1.3, there are two such ways, shown below:



2 Proof of Main Result

We now present a proof of Theorem 1.1. This proof comes from [3] and relies on a previous result from [1] that the Littlewood-Richardson coefficients can be computed via another combinatorial object called honeycombs. Another proof which does not rely on honeycombs can be found in [2]. That proof instead relies on equivariant cohomology of Grassmannians, and also introduces a new "equivariant" puzzle piece.

A brief description of honeycombs is included below, but the reader is encouraged to see [1] for a more detailed description of them and their properties. We do not give a detailed overview of the properties or definition of a honeycomb, but instead only give a few diagrams and important facts in order to give the reader a sense of the objects being described.

The most basic idea of a honeycomb comes from taking a honeycomb-shaped tiling of the plane and cutting out a triangular shape from it, as shown if Figure 2. This gives a shape with finitely many vertices, and the edges which reach the end of the shape are extended out to infinity. This concept can be generalized as shown in Figure 2 so that not every interior edge need be included and the shape cut out need not be an equilateral triangle. Each edge which extends to infinity is given 3 coordinates, based on the three directions the edge can extend in. Thus one of these coordinates will always be a constant and the other two will vary, as shown in Figure 2.

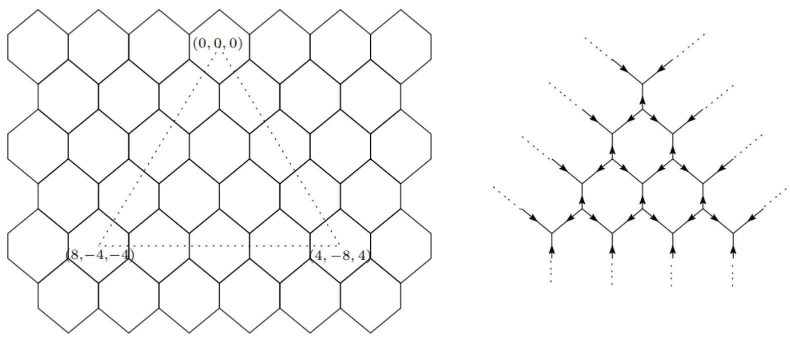


Figure 1: The most basic construction of a honeycomb.

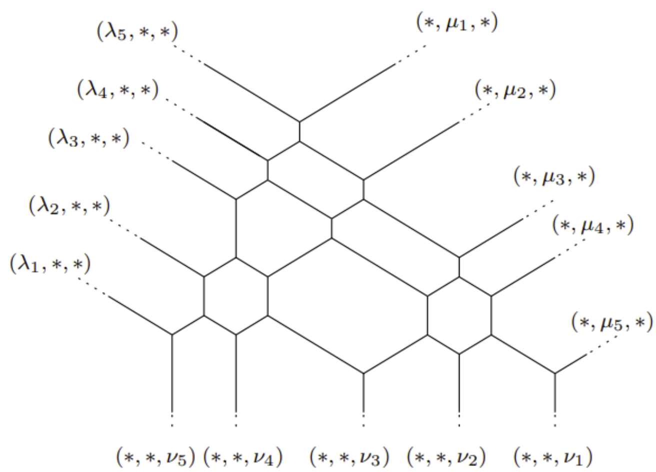


Figure 2: Another example of a honeycomb. Each edge going to infinity has one coordinate which remains constant and two coordinates which vary (marked by *).

The constant coordinates shown in Figure 2 are called the multiplicity of the edge. These mulitplicities form a weakly decreasing sequence, $\lambda_1 \geq \lambda \cdots$, etc.

We will need the following lemma for our proof of Theorem 1.1:

Lemma 2.1. *Let h be a honeycomb with boundary coordinates $(\lambda, \mu, \nu) \in (\mathbb{R}^n)^3$ on the northwest, northeast, and south sides. Then,*

1. *The coordinates of a vertex in h are in $[\lambda_n, \lambda_1], [\mu_n, \mu_1], [\nu_n, \nu_1]$.*
2. *The third coordinate of a vertex in h is in $[-\lambda_1 - \mu_1, -\lambda_n - \mu_n]$.*

Proof. 1. Start at vertex h in the honeycomb and follow a path going northwest whenever possible. Traveling northwest will leave the first coordinate unchanged. If it is not possible to travel northwest, instead travel southwest, which will increase the first coordinate. The result will be exit the honeycomb at an edge with a constant coordinate $\lambda_i \leq \lambda_1$. For the second and third coordinates, rotate everything by 120° or 240° and the proof is the same.

2. It is a fact about honeycombs that the sum of the three coordinates is always 0, so we can find a bound for the third coordinate in terms of the bounds for the first two.

□

Note that the purpose of this lemma is to allow us to describe a convex region which contains all the vertices of the honeycomb, based on the vertices of the edges along the outside.

We can now give the proof of the main result.

Proof of Theorem 1.1. The number of honeycombs with boundary λ, μ, ν was shown in [1] to give $c^{\lambda}_{\mu\nu}$ where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_k)$ as can be seen in Figure 2. Likewise for μ and ν.

To show that puzzles also compute $c^{\lambda}_{\mu\nu}$, we need a bijection between puzzles and honeycombs. We first define the following map from a puzzle to a honeycomb with a three step process (shown in figure 2)

1. Orient the puzzle in the plane with the bottom right corner at the origin, and turn 30° clockwise.

2. At every boundary edge which has a 1, attach a rhombus on the outside of the puzzle, and then continue attaching infinitely many parallel rhombi.

3. Divide the puzzle into regions of adjacent pieces of the same type. Each region of 1-triangles becomes a vertex in the honeycomb, and each rhombus region becomes an edge. The rhombus regions added in the previous step become edges going to infinity. The multiplicity of those edges is the thickness of the original rhombus region.

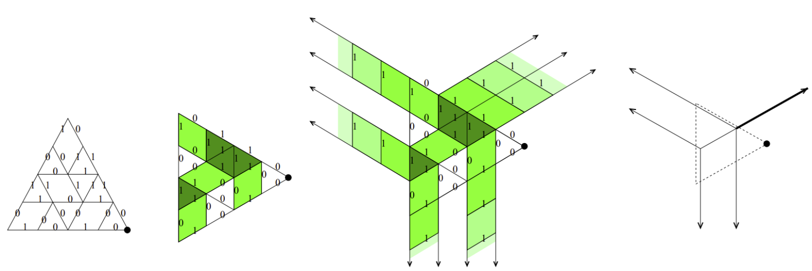


Figure 3: A puzzle being mapped to a honeycomb.

Now we must show that this map is a bijection by constructing the inverse.

Consider a honeycomb which computes $c^{\tau}_{\pi\rho}$. By Lemma 2.1, this honeycomb fits inside the triangle with vertices $(0,0,0), (n-r, 0, r-n), (0, n-r, r-n)$. Each edge of the honeycomb which intersects the triangle becomes a rhombus region. The multiplicity of the edge determines the thickness of the rhombus region. Each vertex becomes a polygon of 1-triangles, where the multiplicities of the edges at each vertex give the lengths of the edges of the polygon. The result is a puzzle with boundary λ, μ, ν. □

3 Further Observations

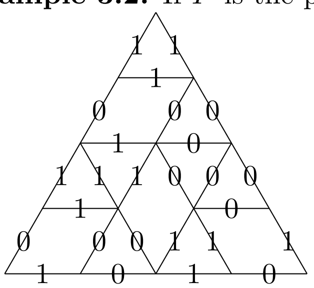
In this section we give some additional interesting facts about puzzles.

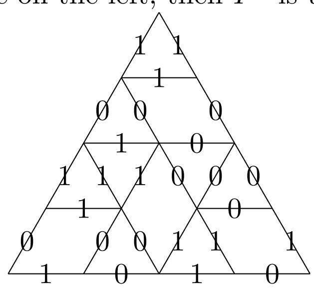
Rotating a puzzle 120° or 240° will always give another puzzle. This can be easily seen by looking at the individual puzzle pieces, each of which gives another puzzle piece when rotated. Note that this does not apply for reflections. It does however apply for the **dual** of a puzzle, defined below:

Definition 3.1. For a puzzle P, the dual of P, denoted P^*,is obtained by reflecting across the center (vertical) line and then swapping 0's and 1's.

To see that the dual of a puzzle is also a puzzle, we can again look at the individual puzzle pieces and observe that reflecting and then swapping 0's and 1's always gives another puzzle piece.

Example 3.2. If P is the puzzle on the left, then P^* is the puzzle on the right:





References

[1] Allen Knutson and Terence Tao. The honeycomb model of gl(n) tensor products i: proof of the saturation conjecture, 1999.

[2] Allen Knutson and Terence Tao. Puzzles and (equivariant) cohomology of grassmannians, 2001.

[3] Allen Knutson, Terence Tao, and Christopher Woodward. The honeycomb model of gl(n) tensor products ii: Puzzles determine facets of the littlewood-richardson cone, 2001.