Tree bijections

Igor Pak, UCLA

IPAM workshop, September 9, 2009

Cayley's problem

Theorem [Cayley, 1889] *The number $T_{n,k}$ of spanning rooted forests on n vertices with k components is $\binom{n-1}{k-1} n^{n-k}$.*

Borchardt (1860), Sylvester (1857) proved *Cayley's formula*:

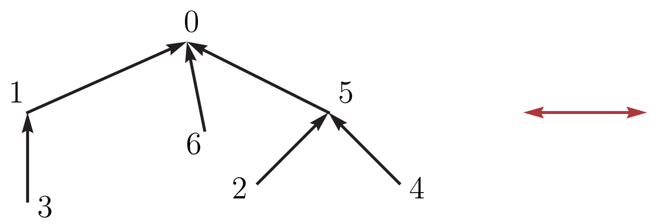
$$T_{n,1} + \ldots + T_{n,n} \;=\; (n+1)^{n-1}.$$

Another version: $T_{n,1} = n^{n-1}$. Cayley proved the theorem by induction.

Proof via *Prüfer's bijection* is perhaps the most straightforward.

Note: A variation on this proof is due to P. Shor (1995). An elegant generalization and a double counting version is due to Pitman (1999).

Prüfer's bijection



$(0, 5, 1, 5, 0)$

Observe: If $k =$ the degree of 0, then 0 appears in the code exactly $k-1$ times. Then, the number of such codes is $T_{n,k} = \binom{n-1}{k-1} n^{n-k}$.

Rényi's problem

Theorem [Scoins, 1962] *The number of spanning trees in a complete bipartite graph* $K_{m,n}$ *is* $m^{n-1}n^{m-1}$.

Theorem [Rényi, 1966] *The number of spanning trees in a complete tripartite graph* $K_{\ell,m,n}$ *is* $(\ell+m)^{n-1}(\ell+n)^{m-1}(m+n)^{\ell-1}(\ell+m+n)$.

Proof idea (in the bipartite case): Observe that Prüfer's code can be split into two subsequences of length $(n-1)$ and $(m-1)$, corresponding to vertices of each part. Now check that the inverse bijection is still well defined.

Note: Rényi extended the result to all multipartite graphs $K_{n_1 \dots n_k}$.
C. & A. Rényi (1970) modified the bijection to work for k-trees.
Greene–Iba (1975) modified the code to work for multidimensional trees.
Kelmans–P.–Postnikov (1997) extended for weighted graphs with part structure.

Parking functions

Theorem [Konheim–Weiss, 1966] *There are* $(n+1)^{n-1}$ *parking functions on* $[n] = \{1, \ldots, n\}$.

A sequence $(a_1, \ldots, a_n)$, $1 \le a_i \le n$ is a *parking function* if

$$\#\{a_i > k\} \le n - k, \quad \text{for all } \ 1 \le k < n.$$

Thus $(1,1)$, $(1,2)$, and $(2,1)$ are parking functions, while $(2,2)$ is not.

Best way to think: A sequence is a parking function if its permutation is majorated by $(1, 2, \ldots, n)$.

Example: For $n = 3$ there are 16 parking functions:
$(1,1,1)$, $\quad (1,1,2) \times 3$, $\quad (1,1,3) \times 3$, $\quad (1,2,2) \times 3$, $\quad (1,2,3) \times 6$.

Note: Schützenberger (1968) found the first bijective proof between parking functions and rooted spanning forests (with no apparent application in mind).

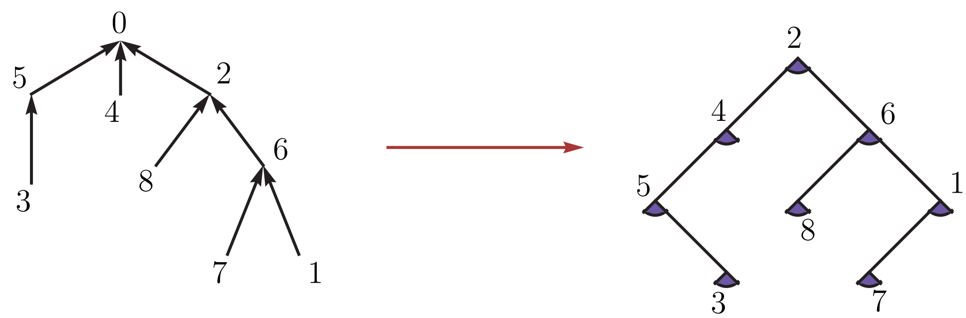
Foata–Riordan problem

Theorem [Foata–Riordan, 1974] *The number of parking functions on $[n]$ with k letters 1 is equal to $\binom{n-1}{k-1} n^{n-k}$.*

Example: For $n = 3$, $k = 2$, you get $\binom{2}{1} 3^1 = 6$ such parking functions.

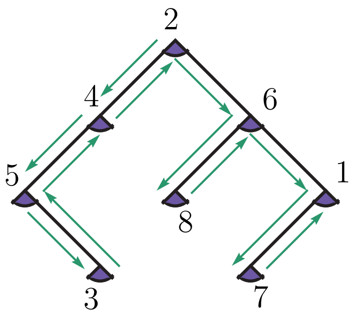
Bijection: rooted forests on $[n] \longrightarrow$ parking functions on $[n]$,
which maps the number of components into the number of letters 1.

First step:



Map: spanning trees $\longrightarrow$ labeled binary trees increasing to the left

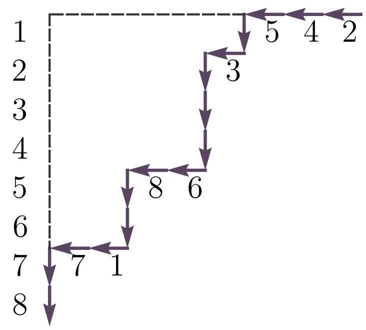
Second step:



$$\longrightarrow \quad (245 * 3 * * * 68 * * 17 * *)$$

Map:
labeled binary trees increasing to the left $\longrightarrow$ labeled Dyck sequences

Last two steps:



$$\longleftarrow \quad (245 * 3 * * * 68 * * 17 * *)$$

$$\longrightarrow \quad \begin{array}{c}(7,1,2,1,1,5,7,5)\\ \hline 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\end{array}$$

Maps: labeled Dyck sequences $\longrightarrow$ labeled lattice paths $\longrightarrow$ parking functions

Shi arrangement

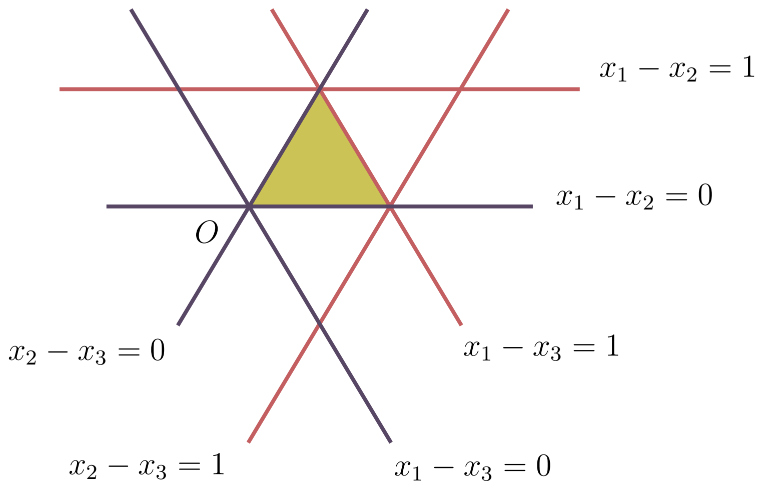
Theorem [Shi, 1986] *The number of regions in $\mathbb{R}^n$ of the hyperplane arrangement Q_n is equal to $(n+1)^{n-1}$.*

Shi arrangement :

$$Q_n := \bigcup_{i<j} \{x_i - x_j = 0\} \bigcup_{i<j} \{x_i - x_j = 1\} \subset \mathbb{R}^n$$

Note: J. Y. Shi proved this by studying certain Kazhdan–Lusztig cells of $\widehat{A}_n$.
He generalized it to other root systems and bounded cells.
Many other proofs and generalizations are known now.
Athanasiadis–Linusson (1999) found a simple bijective proof.

The Shi arrangement Q_3



Here Q_3 has 16 regions.

Stanley's problem

Theorem [P.–Stanley, 1995] *The number of regions of the Shi arrangement* Q_n *at distance* k *from the center region is equal to the number of trees on* $n+1$ *vertices with* $\binom{n}{2} - k$ *inversions.*

Let t be a tree on $\{0, 1, \ldots, n\}$. Fix a root at 0. Vertices $i < j$ form an *inversion* if the path from i to the root goes through j.

There are $n!$ trees with zero inversions. These are *increasing trees.*

Note: Tree inversions were introduced by Mallows–Riordan (1968).
The PS proof is bijective, but does not go the way one would guess.
It is highly non-trivial and still the only known proof of the theorem.

Clue to the proof: the Kreveras theorem

Theorem [Kreveras, 1980] *The number of spanning trees on $n+1$ vertices with k inversions is equal to the number of parking functions on $\{1,\dots,n\}$ with the sum $\binom{n+1}{2} - k$.*

Motivation: Connection to the *Tutte polynomial* of K_n :

$$T_n(1,y) \,=\, \sum_{S\subset E} (y-1)^{|S|-n+1} \,=\, \sum_{t\in K_n} y^{ea(t)},$$

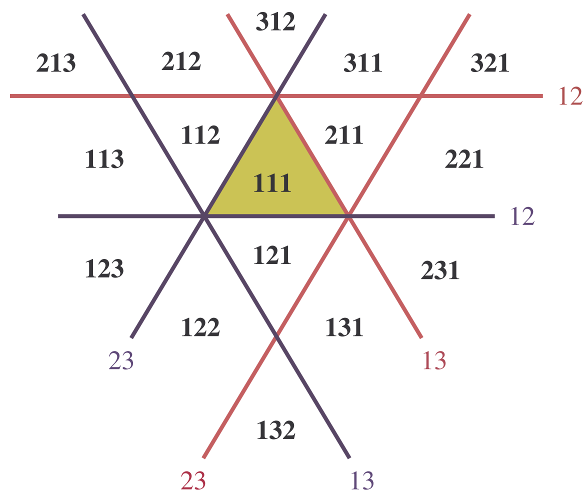
where S are connected subsets of edges in K_n. Kreveras observed:

$$T_n(1,y) \,=\, \sum_{t\in K_n} y^{\mathrm{inv}(t)}\,.$$

Note: Original proof: recurrence relations + Mallows–Riordan identities.
No "nice" bijective proof is known for the Kreveras theorem.
No "nice" bijective proof is known for the change of ordering.

For more on the connection with external activities, see below.

The PS bijection:



Q_3

The rules:

1) when crossing a pink line, increase the *first* of the two coordinates by 1
2) when crossing a blue line, increase the *second* of the two coordinates by 1

Note: Generalizes to other deformed Shi arrangements (Stanley, 1996).
Not to other root systems.

Haiman's problem

Theorem [Haiman, 1994] *The multiplicity of the irreducible representation π_λ of S_n, $\lambda \vdash n$, in a parking representation P_n is equal to*

$$\frac{1}{n+1}\, s_\lambda(1,1,\ldots,1), \quad \leftarrow \ (n+1) \textit{ ones here.}$$

Parking representation: action of S_n on parking function on $[n]$.

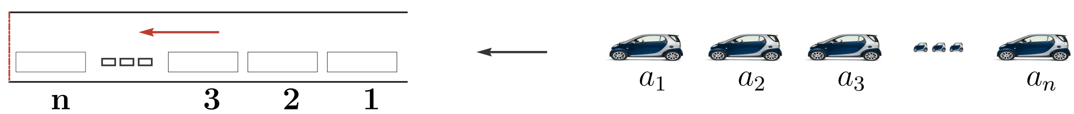
Schur function: $s_\lambda(1,\ldots,1)$ = dimension of the irreducible representation of $GL(N,\mathbb{C})$, corresponding to partition λ.

Note: P_n is a graded representation and appears in the connection with the *diagonal harmonics*, which is naturally bi-graded. It was further studied and refined by Garsia–Haiman (1996, 1998). The connection was eventually proved by Haiman (2001) by using Hilbert schemes and Macdonald polynomials.

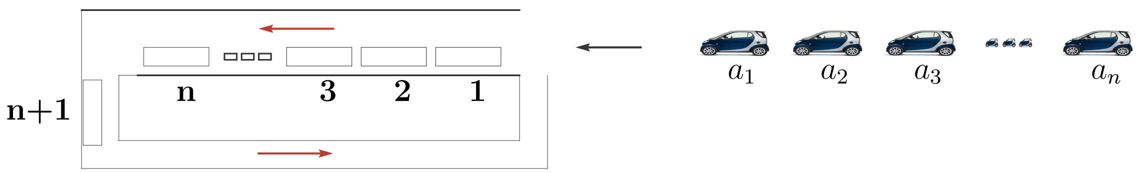
Proof idea (which also explains the name "parking")

Claim: The number of parking functions on $[n] \;=\; \frac{1}{n+1}(n+1)^n$.

Think of $(a_1, \ldots, a_n)$ as preferred parking spots on a one-way street.
Key observation: This is a parking function if and only if all cars can park without backtracking.



Now make the road circular and add one extra parking spot!



Observe that every $[n] \to [n+1]$ function is "parking" in this sense and the only unoccupied parking spot has equal chance to be anywhere.

Recall that we get an old parking function if the new $(n+1)$-th spot is unoccupied. Thus, there are $\frac{1}{n+1}(n+1)^n$ of them. Done!

Note: The basic parking function argument is folklore, sometimes attributed to Pollak (c. 1974). Haiman's proof is based on this argument and algebraic bookkeeping.

Dhar's problem

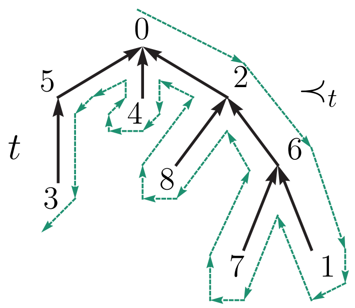
Theorem [Dhar, 1990] *The number G-parking functions is equal to the number of spanning trees in G.*

$G = (V, E)$ is simple graph, $V = \{0, 1, \ldots, n\}$.
$(a_1, \ldots, a_n)$ is a *G-parking function*, if $a_i \in [n]$, and for every $S \subset [n]$ there exists $i \in S$, such that $\#\{j \in V - S \mid (i, j) \in E\} \geq a_i$.

For $G = K_{n+1}$ these are the usual parking functions.

Note: The G-parking functions originate independently from several diverse sources:
(1) Bak–Tang–Wiesenfeld (1987), Dhar (1990), Gabrielov (1993), sandpile models
(2) Björner–Lovász–Shor (1991), chip-firing games
(3) Bacher–de la Harpe–Nagnibeda (1997), integral flows and cuts on finite graphs

Depth-first search order on trees:



$$\longrightarrow \qquad \frac{(2,6,1,7,8,4,5,3)}{(0,1,2,2,1,0,0,6)}$$

New bijection: spanning trees in $K_{n+1} \longrightarrow$ parking functions on $[n]$.

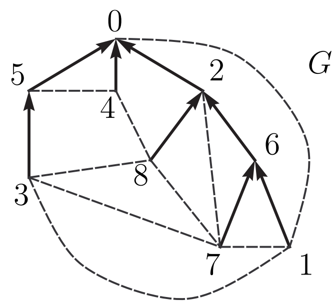
Rule: If $i \to j$ in a tree, let $a_i = \#\{\ell \mid \ell \prec_t j, 0 \le \ell \le n\} + 1$.

In the example, the corresponding parking function is $(3,1,7,1,1,2,3,2)$.

Note: This bijection is due to Françon (1975).

The Breadth-first search and certain other orderings also work.

General case:



$$\longrightarrow \quad \frac{(2,6,1,7,8,4,5,3)}{(0,0,1,1,0,0,0,3)}$$

Rule: If $i \to j$ in a tree, let $a_i = \#\{\ell \mid \ell \prec_t j, (i,\ell) \in E\} + 1$.

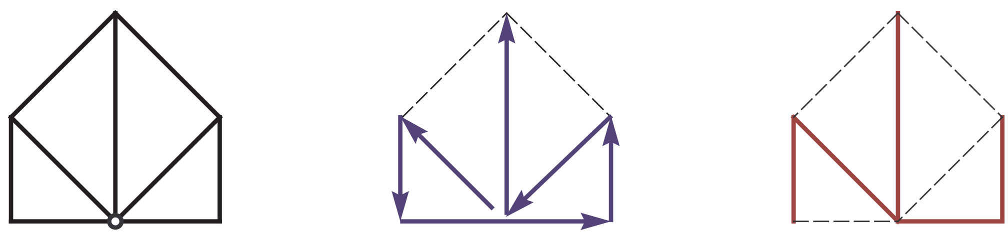
In the example, the corresponding G-parking function is $(2,1,4,1,1,1,2,1)$.

Note: In this form the bijection is due to Dhar–Majumdar (1992). Dhar used a version of it to give a linear time test whether a sequence is a G-parking function.
For directed graphs an analogue was given by Chebikin–Pylyavskyy (2005).
The analogue of Kreveras connection to Tutte polynomial was discovered by Biggs.
This was proved by Merino López (1997), and via bijection by Cori–Le Borgne (2003).

CS problem:

How to generate a uniform random spanning tree in a given graph?

Algorithm [Aldous, 1990; Broder, 1989]
Fix a vertex v in a given simple graph G.
Run a nearest neighbor random walk on G for $T =$ cover time steps.
Output a tree consisting of the edges with first visits to vertices.



Note: For example, in expanders the cover time $T = O(n \log n)$, which gives a nearly linear algorithm for the random tree generation. Broder's proof is elementary. Aldous's proof uses a delicate MC argument. He then obtains new bounds on the diameter of random spanning trees and on the distribution of the number of leaves.

What happens for a complete graph K_n

Algorithm [Aldous, 1990]
Let $Z_i \in [n]$ be uniform and independent r.v., for all $i = 2, \ldots, n$.
Connect $i \to \min\{i-1, Z_i\}$.
Relabel the vertices according to a random permutation $\sigma \in S_n$.
Output the resulting random tree.

Theorem [Aldous]
This algorithm outputs a uniform spanning tree in K_n.

Question: Why?

Loop-erased random walk

Algorithm [D. Wilson, 1996]

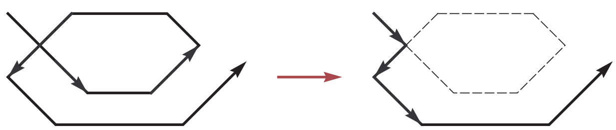
Start at 2. Walk until 1 is hit, erasing loops as they are created.

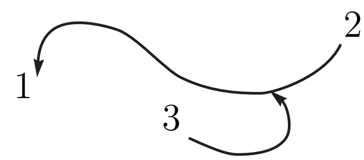
Fix the resulting $2 \to 1$ path.

Start at 3. Walk until the $2 \to 1$ path is hit, erasing loops.

Fix the resulting $3 \to (2 \to 1)$ path.

Continue in this manner until a uniform spanning tree is obtained.





Note: Wilson's algorithm is at least as fast as the cover time; sometimes much faster.
Brought an explosion of work on random spanning forests in infinite transitive graphs (Lawler, Lyons, Pemantle, Peres, Schramm, etc.)
Analyzed by P. Marchal (2000) based on the classical Viennot bijection.