Multi-parameter hook formula for labelled trees

Valentin Féray
joint work with Ian P. Goulden (Waterloo)
and Alain Lascoux (Marne-La-Vallée)

Institut für Mathematik, Universität Zürich

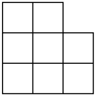
Séminaire Philippe Flajolet,
IHP, Paris, October 3rd, 2013

Outline of the talk

1. What is a hook formula?
2. Main result and specializations
3. A combinatorial proof of our hook formula: splicing trees

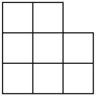
Frame-Robinson-Thrall formula (1954) for counting tableaux

Fix a Young diagram λ with n boxes.

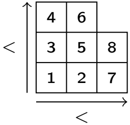


Frame-Robinson-Thrall formula (1954) for counting tableaux

Fix a Young diagram λ with n boxes.



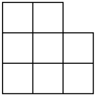
Then the number of standard Young tableaux



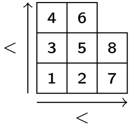
is given by

Frame-Robinson-Thrall formula (1954) for counting tableaux

Fix a Young diagram λ with n boxes.



Then the number of standard Young tableaux



is given by

$$\frac{n!}{\prod_{\square \in \lambda} h_\square}.$$

$h_\square$: hook-length of the box $\square$, *i.e.* number of boxes at its right in the same row or above it in the same column.

In our example: the hook-lengths are

2	1	
4	3	1
5	4	2

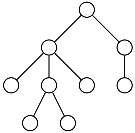
so there are

$8!/(5*4*4*3*2*2) = 42$ standard Young tableaux of shape λ.

Knuth formula for increasing trees (1973)

The same kind of formula holds for trees!

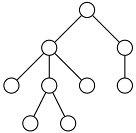
Fix a Tree T with n nodes.



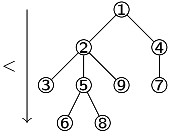
Knuth formula for increasing trees (1973)

The same kind of formula holds for trees!

Fix a Tree T with n nodes.



Then the number of increasing labellings of this tree

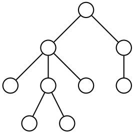


is given by

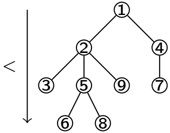
Knuth formula for increasing trees (1973)

The same kind of formula holds for trees!

Fix a Tree T with n nodes.



Then the number of increasing labellings of this tree

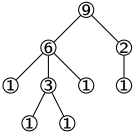


is given by

$$\frac{n!}{\prod_{\circ \in V(T)} h_\circ}.$$

$h_\circ$: hook-length of the vertex $\circ$, *i.e.* the number of vertices in the subtree of T rooted in $\circ$.

In our example: the hook-lengths are



so there are

$9!/(9 * 6 * 3 * 2) = 1120$ increasing labellings of T.

Hook summation formulas

But these objects are in bijection with permutations.

- By Robinson-Schensted algorithm, pairs of standard Young tableaux of the same shape are in bijection with permutations, so

$$\sum_{\lambda \vdash n} \left(\frac{n!}{\prod_{\square \in \lambda} h_\square} \right)^2 = n!.$$

Hook summation formulas

But these objects are in bijection with permutations.

- By Robinson-Schensted algorithm, pairs of standard Young tableaux of the same shape are in bijection with permutations, so

$$\sum_{\lambda \vdash n} \left(\frac{n!}{\prod_{\square \in \lambda} h_\square} \right)^2 = n!.$$

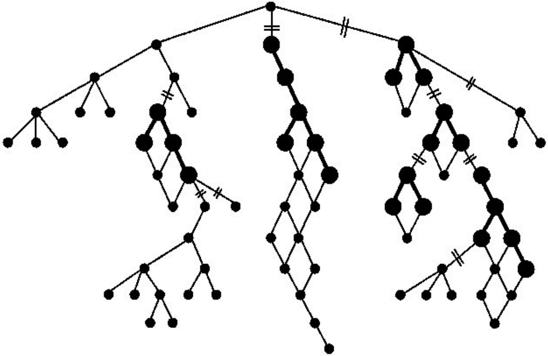
- By binary search tree algorithm, increasing labellings of binary trees are in bijection with permutations, so

$$\sum_{T \text{ binary tree}} \frac{n!}{\prod_{\circ \in V_T} h_\circ} = n!$$

These formulas are called hook summation formulas.

A large amount of work around these hook formulas

- formulas for other objects than trees or Young diagrams: in particular, *d*-complete posets that include both.



© R. Proctor

A large amount of work around these hook formulas

- formulas for other objects than trees or Young diagrams: in particular, d-complete posets that include both.
- in summation formulas, one can replace $1/h_\square^2$ or $1/h_\circ$ by more involved expressions such that the sum is still simple.

Example (Postnikov formula)

$$\sum_{\substack{T \text{ binary} \\ \text{tree of size } n}} \prod_{v \in T} \left(x + \frac{1}{h_T(v)} \right) = \frac{1}{(n+1)!} \prod_{i=1}^{n-1} \big((n+1+i)x + n + 1 - i \big).$$

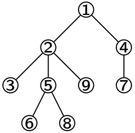
A large amount of work around these hook formulas

- formulas for other objects than trees or Young diagrams: in particular, d-complete posets that include both.
- in summation formulas, one can replace $1/h_{\square}^2$ or $1/h_{\circ}$ by more involved expressions such that the sum is still simple.
- interpretations in combinatorial Hopf algebra theory, in convex geometry, in commutative algebra.
- ...

Main result

A hook summation formula over labelled increasing tree with n nodes.

A labelled increasing tree T



Children of a given vertex are not ordered. By convention, we draw them in increasing order from left to right.

⚠ in our formula, we sum over labelled trees.

Main result

A hook summation formula over labelled increasing tree with n nodes.

Theorem (FGL, 2013)

Let $(x_i)_{1\leq i\leq n}$ and $(y_{i,j})_{1\leq i\leq j\leq n}$ be formal parameters.

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

$f_i(T)$: parent of i in T;
$\mathfrak{h}_i(T)$: vertex set of the sub-tree of T rooted in i.

Example :

$$\text{weight}\left(\begin{array}{c} ① \\ | \\ ② \\ | \\ ③ \end{array} \right) = x_1(y_{2,2}+y_{2,3})x_2 y_{3,3}$$

Main result

A hook summation formula over labelled increasing tree with n nodes.

Theorem (FGL, 2013)

Let $(x_i)_{1 \le i \le n}$ and $(y_{i,j})_{1 \le i \le j \le n}$ be formal parameters.

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

$f_i(T)$: parent of i in T;
$\mathfrak{h}_i(T)$: vertex set of the subtree of T rooted in i.

Example :

$$\text{weight}\left(\begin{array}{c} \textcircled{1} \\ | \\ \textcircled{2} \\ | \\ \textcircled{3} \end{array} \right) = x_1(y_{2,2} + y_{2,3}) x_2 y_{3,3}$$

A specialization ($y_{i,j} = x_j + \delta_{i,j} - 1$) appeared in representation theory of symmetric groups.

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
With this specialization, the weight of a tree is

$$\text{weight}(T) = \prod_{v} (y + z \cdot |\mathfrak{h}_v(T)|),$$

where the product runs over non-root vertices.

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
With this specialization, the weight of a tree is

$$\mathrm{weight}(T) = \prod_v \left(y + z \cdot |\mathfrak{h}_v(T)|\right),$$

where the product runs over non-root vertices.
It does not depend on the labelling of T!

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
With this specialization, the weight of a tree is

$$\mathrm{weight}(T) = \prod_{v} (y + z \cdot |\mathfrak{h}_v(T)|),$$

where the product runs over non-root vertices.
It does not depend on the labelling of T! Hence

$$\mathrm{LHS} = \sum_{\substack{T \text{ labelled} \\ \text{tree}}} \mathrm{weight}(T) = \sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \#\{\text{labellings}\}\, \mathrm{weight}(U)$$

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
With this specialization, the weight of a tree is

$$\mathrm{weight}(T) = \prod_v \big(y + z \cdot |\mathfrak{h}_v(T)|\big),$$

where the product runs over non-root vertices.
It does not depend on the labelling of T! Hence

$$\begin{aligned}
\mathrm{LHS} &= \sum_{\substack{T \text{ labelled} \\ \text{tree}}} \mathrm{weight}(T) = \sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \#\{\text{labellings}\}\, \mathrm{weight}(U) \\
&= \sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \frac{n!}{\prod_v |\mathfrak{h}_v(T)|} \prod_{v \text{ non-root}} \big(y + z \cdot |\mathfrak{h}_v(T)|\big) \\
&= (n-1)! \sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \prod_{v \text{ non-root}} \left(\frac{y}{|\mathfrak{h}_v(T)|} + z\right).
\end{aligned}$$

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
Finally, we get

$$\sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \prod_{v \text{ non-root}} \left(\frac{y}{|\mathfrak{h}_v(T)|} + z \right) = \frac{1}{n!} \prod_{i=2}^{n} \left(i \cdot y + (n-1) \cdot z \right)$$

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
Finally, we get

$$\sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \prod_{v \text{ non-root}} \left(\frac{y}{|\mathfrak{h}_v(T)|} + z \right) = \frac{1}{n!} \prod_{i=2}^{n} \left(i \cdot y + (n-1) \cdot z \right)$$

Looks a lot like Postnikov's formula except that the sum runs over trees with any arity (not binary trees).

An interesting specialization

Set $x_i = 1$, $y_{i,i} = y$ and $y_{i,j} = z$ for $i \neq j$.
Finally, we get

$$\sum_{\substack{U \text{ unlabelled} \\ \text{tree}}} \prod_{v \text{ non-root}} \left(\frac{y}{|\mathfrak{h}_v(T)|} + z \right) = \frac{1}{n!} \prod_{i=2}^{n} \big(i \cdot y + (n-1) \cdot z \big)$$

Looks a lot like Postnikov's formula except that the sum runs over trees with any arity (not binary trees).

Summing over labelled tree is natural to get a multi-parameter generalization!

Question

Is there a formula similar to our main result with a sum over labelled binary tree?

Another interesting specialization: recovering Cayley formula

Set $y_{i,j} = x_j$ for every $i \leq j$. Then

$$\mathrm{RHS} = x_1 \dots x_n \left(\sum_{j=1}^{n} x_j \right)^{n-2}.$$

Another interesting specialization: recovering Cayley formula

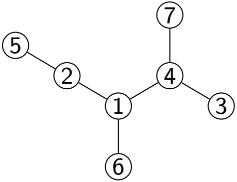
Set $y_{i,j} = x_j$ for every $i \le j$. Then

$$\mathrm{RHS} = x_1 \dots x_n \left(\sum_{j=1}^{n} x_j \right)^{n-2}.$$

We would like to show that

$$\mathrm{LHS} = \sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} x_j \right) \right] = \sum_{T \text{ Cayley tree}} x_1^{\deg_1(T)} \dots x_n^{\deg_n(T)}.$$

Reminder:
A Cayley tree
(no root, no plane embedding)



Another interesting specialization: recovering Cayley formula

Let X a subset of $[n]$. We define:

$$\mathrm{LHS}(X) = \sum_{\substack{T \text{ incresing tree} \\ \text{with label set } X}} \left[\prod_{i \in X \setminus \{\min(X)\}} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} x_j \right) \right]$$

$$\mathrm{Cay}(X) = \sum_{\substack{T \text{ Cayley tree} \\ \text{with label set } X}} x_1^{\deg_1(T)} \dots x_n^{\deg_n(T)}.$$

Another interesting specialization: recovering Cayley formula

Let X a subset of $[n]$. We define:

$$\mathrm{LHS}(X) = \sum_{\substack{T \text{ incresing tree} \\ \text{with label set } X}} \left[\prod_{i \in X \setminus \{\min(X)\}} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} x_j \right) \right]$$

$$\mathrm{Cay}(X) = \sum_{\substack{T \text{ Cayley tree} \\ \text{with label set } X}} x_1^{\deg_1(T)} \dots x_n^{\deg_n(T)}.$$

Proof that $\mathrm{LHS}(X) = \mathrm{Cay}(X)$

Both satisfy the same induction (and coincide for $|X| = 2$)

$$F(X) = \sum_d x_{\min(X)}^d \sum_{X_1 \sqcup \dots \sqcup X_d = X \setminus \{\min(X)\}} \left(\prod_{i=1}^r F(X_i) \sum_{v \in X_i} x_v \right).$$

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Reminder: this is our main result.

We would like a combinatorial formulation.

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
In the left hand-side:

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
In the left hand-side:

- To contribute, a tree must fulfill:

$$2 \leq_T 7,\ 3 \leq_T 4, 5 \leq_T 7$$

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
In the left hand-side:

- To contribute, a tree must fulfill:

$$2 \leq_T 7,\ 3 \leq_T 4, 5 \leq_T 7$$

  This implies also $2 \leq_T 5$. Because we are using trees!

Towards a combinatorial formulation

$$\sum_{T} \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
In the left hand-side:

- To contribute, a tree must fulfill:

$$2 \leq_T 7,\ 3 \leq_T 4, 5 \leq_T 7$$

 This implies also $2 \leq_T 5$. In general, the monomial M_y defines a set-partition π of $\{2, \dots, n\}$ and elements from the same part must be in the same path from the root to a leaf.

In the example, $\pi = \{\{2, 5, 7\}, \{3, 4\}, \{6\}\}$

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
In the left hand-side:

- To contribute, a tree must fulfill:

  $$2 \leq_T 7,\ 3 \leq_T 4, 5 \leq_T 7$$

  This implies also $2 \leq_T 5$. In general, the monomial M_y defines a set-partition π of $\{2, \dots, n\}$ and elements from the same part must be in the same path from the root to a leaf.

  In the example, $\pi = \{\{2,5,7\},\{3,4\},\{6\}\}$

- The contribution of a tree T is $\prod_i x_i^{\deg_T(i)}$.

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^n x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$. Finally,

$$[M_y]\,\mathrm{LHS} = \sum_T \prod_i x_i^{\deg_T(i)},$$

where the sum runs over π-compatible trees.

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
Consider now the right-hand side

$$[M_y]\,\mathrm{RHS} = x_1\, x_2\, x_3 \left(\sum_{j=1}^{4} x_j \right) x_5 \left(\sum_{j=1}^{6} x_j \right).$$

Towards a combinatorial formulation

$$\sum_T \left[\prod_{i=2}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$

Consider, for instance, the coefficient of $M_y := y_{2,7}\, y_{3,4}\, y_{4,4}\, y_{5,7}\, y_{6,6}\, y_{7,7}$.
Consider now the right-hand side

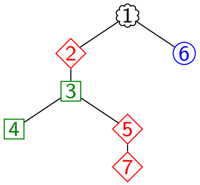
$$[M_y]\,\mathrm{RHS} = x_1\, x_2\, x_3 \left(\sum_{j=1}^{4} x_j \right) x_5 \left(\sum_{j=1}^{6} x_j \right).$$

In general,

$$[M_y]\,\mathrm{RHS} = x_1 \cdot \left[\prod_{\substack{i \text{ not max} \\ \text{in its part}}} x_i \right] \cdot \left[\prod_{\substack{i \text{ max} \\ \text{in its part} \\ i \neq n}} \left(\sum_{j=1}^{i} x_j \right) \right].$$

Combinatorial reformulation of the main theorem

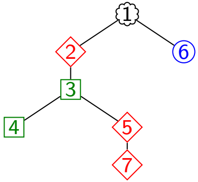
Fix a set-partition of $\{2, \dots, n\}$ (in the example $\pi = \{\{2,5,7\},\{3,4\},\{6\}\}$). One has to find a bijection between



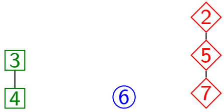
increasing trees T such that, for any two elements in the same part, one is the ancestor of the other.

Combinatorial reformulation of the main theorem

Fix a set-partition of $\{2, \dots, n\}$ (in the example $\pi = \{\{2,5,7\},\{3,4\},\{6\}\}$). One has to find a bijection between



increasing trees T such that, for any two elements in the same part, one is the ancestor of the other.



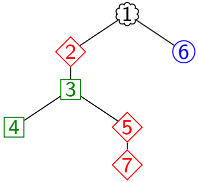
$a_{\square} = 1 \quad a_{\circ} = 3 \quad \emptyset$

a number for each part (except the one containing n) less or equal than the maximum of the part (called *anchor point*)

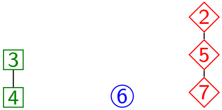
$$a_{\square} \leq 4, \;\; a_{\circ} \leq 6.$$

Combinatorial reformulation of the main theorem

Fix a set-partition of $\{2, \dots, n\}$ (in the example $\pi = \{\{2,5,7\},\{3,4\},\{6\}\}$). One has to find a bijection between



increasing trees T such that, for any two elements in the same part, one is the ancestor of the other.



$a_{\square} = 1$ $\quad a_{\circ} = 3$ $\quad \emptyset$

a number for each part (except the one containing n) less or equal than the maximum of the part (called *anchor point*)

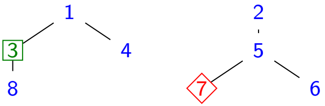
$$a_{\square} \leq 4, \;\; a_{\circ} \leq 6.$$

which respects the degree:

$$\deg_{\text{left}}(i) = \deg_{\text{right}}(i) + |a^{-1}(i)| + \delta_{i,1}.$$

Elementary splicing

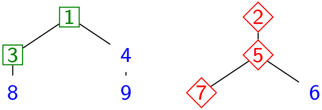
Let T_1 and T_2 with marked vertices v_1 and v_2. Assume $v_1 < v_2$.



In the example $v_1 = 3$, $v_2 = 7$.

Elementary splicing

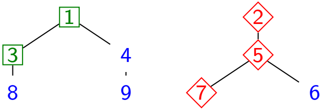
Let T_1 and T_2 with marked vertices v_1 and v_2. Assume $v_1 < v_2$.



Consider the chain from the root to v_1 (resp. v_2).

Elementary splicing

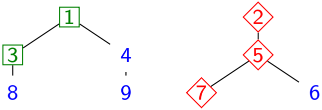
Let T_1 and T_2 with marked vertices v_1 and v_2. Assume $v_1 < v_2$.



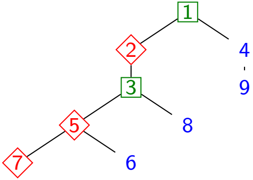
Consider the chain from the root to v_1 (resp. v_2).
These two chains can be merged in an increasing chain in a unique way.

Elementary splicing

Let T_1 and T_2 with marked vertices v_1 and v_2. Assume $v_1 < v_2$.

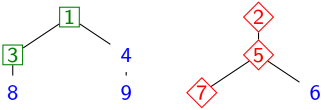


Consider the chain from the root to v_1 (resp. v_2).
These two chains can be merged in an increasing chain in a unique way.
We add other vertices with the same parent than in the original trees:

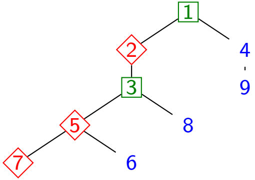


Elementary splicing

Let T_1 and T_2 with marked vertices v_1 and v_2. Assume $v_1 < v_2$.

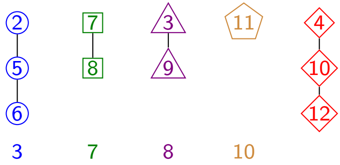


Consider the chain from the root to v_1 (resp. v_2).
These two chains can be merged in an increasing chain in a unique way.
We add other vertices with the same parent than in the original trees:



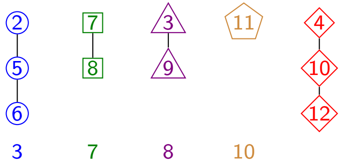
Obs. only the degree of v_1 has increase by 1, other degrees are unchanged.

The bijection on an example



Start with the set of chains above with anchor points.

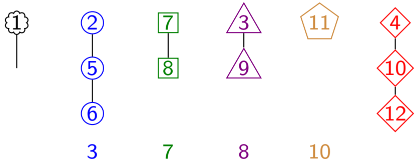
The bijection on an example



Start with the set of chains above with anchor points.
Step 0: we add a root labeled 1 with a free edge to the list.

The free edge symbolizes that we must increase the degree of the corresponding vertex of 1 during the construction.

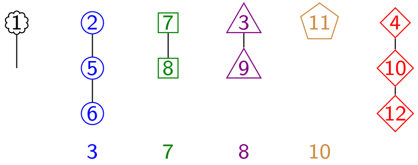
The bijection on an example



Start with the set of chains above with anchor points.
Step 0: we add a root labeled 1 with a free edge to the list.

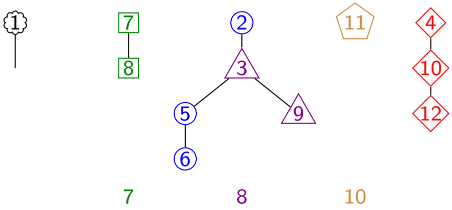
The free edge symbolizes that we must increase the degree of the corresponding vertex of 1 during the construction.

The bijection on an example



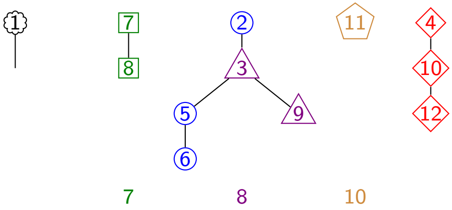
We will splice successively the chains together (always with v_1 a vertex with a free edge, v_2 the max of its tree).
First step: we add a free edge to 3 and splice $2, 5, 6$ with $3, 9$ (*external splice*).

The bijection on an example



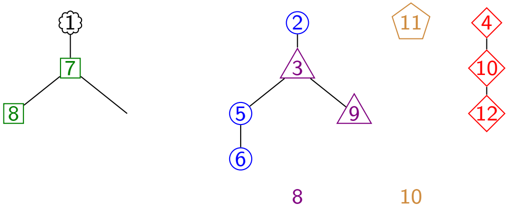
We will splice successively the chains together (always with v_1 a vertex with a free edge, v_2 the max of its tree).
First step: we add a free edge to 3 and splice $2, 5, 6$ with $3, 9$ (*external splice*).

The bijection on an example



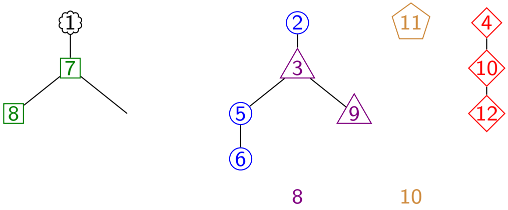
Second step: 7 is in the component we must splice. Thus, we splice 7, 8 on the free edge and add a free edge to 7 (*internal splice*).

The bijection on an example



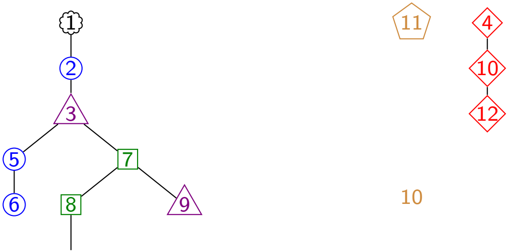
Second step: 7 is in the component we must splice. Thus, we splice $7, 8$ on the free edge and add a free edge to 7 (*internal splice*).

The bijection on an example



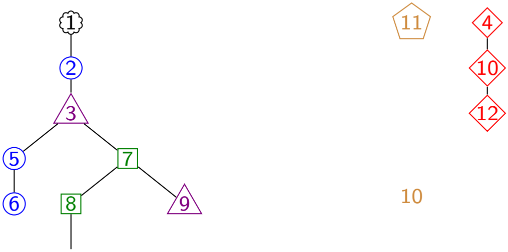
Third step: 8 is in the root component ⇒ again an internal splice. We splice the tree $2, 3, 5, 6, 9$ onto the free edge and add a free edge to 8.

The bijection on an example



Third step: 8 is in the root component ⇒ again an internal splice. We splice the tree $2, 3, 5, 6, 9$ onto the free edge and add a free edge to 8.

The bijection on an example

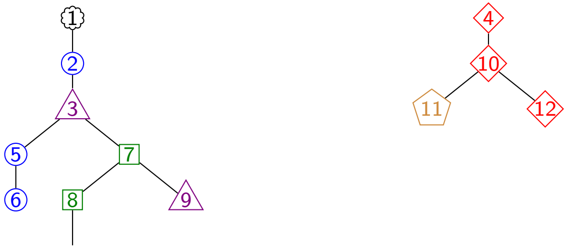


Fourth step: an external splice. We add a free edge to 10 and splice 11 onto it.

The bijection on an example

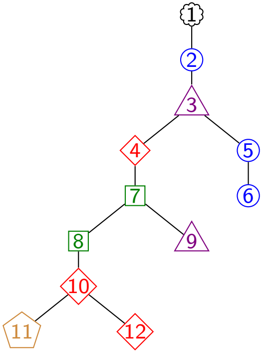
Fourth step: an external splice. We add a free edge to 10 and splice 11 onto it.

The bijection on an example



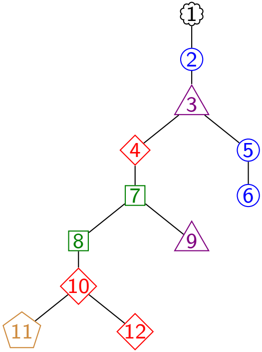
Last step: we splice the tree containing the maximum onto the free edge.

The bijection on an example



Last step: we splice the tree containing the maximum onto the free edge.

The bijection on an example



Here is the resulting partitioned tree.
The degree condition is fulfilled by construction.

Summary and conclusion

Construction by successive splicings:

- if the anchor point is in the component we want to splice or in the root component, we splice onto the free edge and add an edge to the anchor point (internal splicing).
- if the anchor point is in another component, we add a free edge to the anchor point and splice the tree on this free edge (external splicing).

Summary and conclusion

Construction by successive splicings:

- if the anchor point is in the component we want to splice or in the root component, we splice onto the free edge and add an edge to the anchor point (internal splicing).
- if the anchor point is in another component, we add a free edge to the anchor point and splice the tree on this free edge (external splicing).

Theorem (FGL, 2013)

The described procedure defines a bijection.

Summary and conclusion

Construction by successive splicings.

- if the anchor point is in the component we want to splice or in the root component, we splice onto the free edge and add an edge to the anchor point (internal splicing).
- if the anchor point is in another component, we add a free edge to the anchor point and splice the tree on this free edge (external splicing).

Theorem (FGL, 2013)

The described procedure defines a bijection.

Corollary (FGL, 2013)

$$\sum_T \left[\prod_{i=1}^{n} x_{f_i(T)} \left(\sum_{j \in \mathfrak{h}_i(T)} y_{i,j} \right) \right] = x_1 y_{n,n} \prod_{i=2}^{n-1} \left(y_{i,i} \sum_{j=1}^{i} x_j + x_i \sum_{j=i+1}^{n} y_{i,j} \right).$$