Combinatorics of non-ambiguous trees

Jean-Christophe Aval, Adrien Boussicault, Mathilde Bouvel, Matteo Silimbani [*]

Abstract

This article investigates combinatorial properties of non-ambiguous trees. These objects we define may be seen either as binary trees drawn on a grid with some constraints, or as a subset of the tree-like tableaux previously defined by Aval, Boussicault and Nadeau. The enumeration of non-ambiguous trees satisfying some additional constraints allows us to give elegant combinatorial proofs of identities due to Carlitz, and to Ehrenborg and Steingrímsson. We also provide a hook formula to count the number of non-ambiguous trees with a given underlying tree. Finally, we use non-ambiguous trees to describe a very natural bijection between parallelogram polyominoes and binary trees.

1   Introduction

It is well known that the Catalan numbers $C_n = \frac{1}{n+1}\binom{2n}{n}$ enumerate many combinatorial objects, such as binary trees and parallelogram polyominoes. Several bijective proofs in the literature show that parallelogram polyominoes are enumerated by Catalan numbers, the two most classical being Delest-Viennot's bijection with Dyck paths [DV84] and Viennot's bijection with bicolored Motzkin paths [DV84].

In this paper we demonstrate a bijection – which we believe is more natural – between binary trees and parallelogram polyominoes. In some sense, we show that parallelogram polyominoes may be seen as two-dimensional drawings of binary trees. This point of view gives rise to a new family of objects – we call them *non-ambiguous trees* – which are particular compact embeddings of binary trees in a grid.

The tree structure of these objects leads to a hook formula for the number of non-ambiguous trees with a given underlying tree. Unlike the classical hook formula for trees due to Knuth (see [Knu98], §5.1.4, Exercise 20), this one is defined on the edges of the tree.

Non-ambiguous trees are in bijection with permutations such that all their (strict) excedances stand at the beginning of the permutation word. Ehrenborg and Steingrímsson in [ES00] give a closed formula (involving Stirling numbers of the second kind) for the number of such permutations. We show that this formula can be easily proved using non-ambiguous trees and a variation of the insertion algorithm for tree-like tableaux introduced in [ABN11]. Indeed, non-ambiguous trees can also be seen as a subclass of tree-like tableaux, objects defined in [ABN11] and that are in bijection with permutation tableaux [SW07] or alternative tableaux [Nad11, Vie07].

A particular subclass of non-ambiguous trees leads to unexpected combinatorial interpretations. We study complete non-ambiguous trees, defined as non-ambiguous trees such that their underlying binary tree is complete, and show that their enumerating sequence is related to the formal power series of the logarithm of the Bessel function of order 0. This gives rise to new combinatorial interpretations of some identities due to Carlitz [Car63], and to the proof of a related identity involving Catalan numbers, which had been conjectured by P. Hanna.

The paper is organized as follows: in Section 2 we define non-ambiguous trees. Then, in Section 3 we give the enumeration of non-ambiguous trees satisfying certain constraints: those

---

contained in a given rectangular box, and those with a fixed underlying tree. Section 4 introduces the family of complete non-ambiguous trees, studies the relations between this family and the Bessel function, and proves combinatorial identities involving Catalan numbers and the sequence enumerating complete non-ambiguous trees. In Section 5 we describe our new bijection between binary trees and parallelogram polyominoes. To conclude, we present in Section 6 some perspectives related to our study.

2 Definitions and notations

In this paper, trees are embedded in a bidimensional grid $\mathbb{N} \times \mathbb{N}$. The grid is not oriented as usual: the x-axis has south-west orientation, and the y-axis has south-east orientation, as shown on Figure 1.
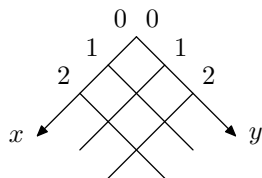

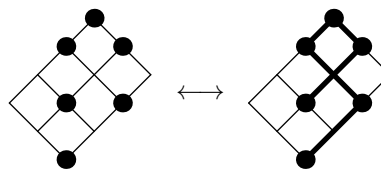
Figure 1: The underlying grid for non-ambiguous trees

Figure 2: The edges of a non-ambiguous tree are not necessary

Every x-oriented (resp. y-oriented) line will be called column (resp. row). Each column (resp. row) on this grid is numbered with an integer corresponding to its y (resp. x) coordinate. A vertex v located on the intersection of two lines has the coordinate representation: $(X(v), Y(v))$.

A non-ambiguous tree may be seen as a binary tree embedded in the grid in such a way that its edges may be recovered from the embedding of its vertices in the grid (Figure 2). In other words, the vertices determine the tree without ambiguity, whence the name of these objects.

Formally, a *non-ambiguous tree* of size n is a set A of n points $(x, y) \in \mathbb{N} \times \mathbb{N}$ such that:

1. $(0,0) \in A$; we call this point the *root* of A;

2. given a non-root point $p \in A$, there exists one point $q \in A$ such that $Y(q) < Y(p)$ and $X(q) = X(p)$, or one point $r \in A$ such that $X(r) < X(p)$, $Y(r) = Y(p)$, but not both (which means that the pattern $\vee$ is avoided);

3. there is no empty line between two given points: if there exists a point $p \in A$ such that $X(p) = x$ (resp. $Y(p) = y$), then for every $x' < x$ (resp. $y' < y$) there exists $q \in A$ such that $X(q) = x'$ (resp. $Y(q) = y'$).

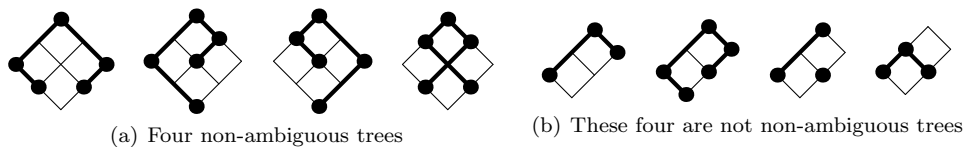Figure 3 shows some examples and counterexamples of non-ambiguous trees.

(a) Four non-ambiguous trees (b) These four are not non-ambiguous trees

Figure 3: Some examples and counterexamples of non-ambiguous trees

It is straightforward that a non-ambiguous tree A has a tree structure: except for the root, every point $p \in A$ has a unique parent, which is the nearest point q preceding p in the same row (resp. column). In this case, we will say that p is the *right child* (resp. *left child*) of q. In this paper, we orient every edge of a tree from the root to the leaves. We shall denote by $\mathsf{T}(A)$ the underlying binary tree associated to A.

Figure 4 shows all the non-ambiguous trees of size 4, grouping inside a rectangle those having the same underlying binary tree.
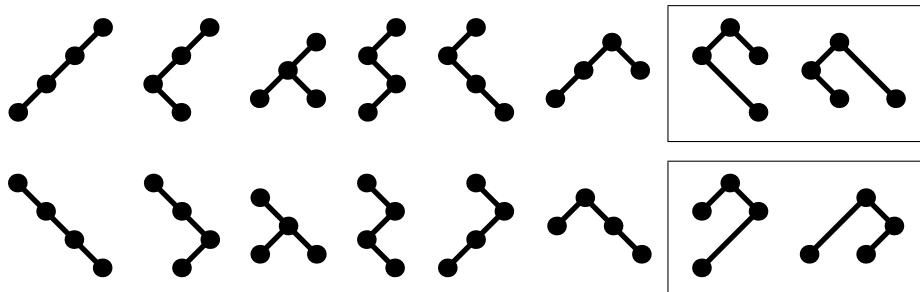

Figure 4: The 16 non-ambiguous trees of size 4

Remark 1. *A tree-like tableau [ABN11] of size n is a set of n points placed in the boxes of a Ferrers diagram such that conditions 1, 2, 3 defining non-ambiguous trees are satisfied. Figure 5 shows an example of a tree-like tableau of size 7. It should be clear that non-ambiguous trees are in bijection with tree-like tableaux with rectangular shape.*
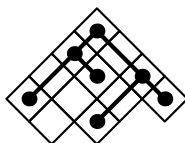

Figure 5: A tree-like tableau

3 Enumeration of non-ambiguous trees

Eventhough non-ambiguous trees are new combinatorial objects, their enumeration has been previously studied. Indeed, non-ambiguous trees of size n are in bijection with permutations of size n with all their strict excedances at the beginning, whose enumeration has been studied by Ehrenborg and Steingrímsson in [ES00]. The size preserving bijection (that we do not detail here) between these two families of combinatorial objects is a consequence of Lemma 5 in [SW07] and of results proved in [ABN13]. The sequence $(a_n)_{n \geq 1}$ counting the number of non-ambiguous tree of size n is referenced in [Slo] as `A136127` $= [1, 2, 5, 16, 63, 294, 1585, 9692, \dots]$, but no simple formula is known. In this section, we provide enumerative formulas for non-ambiguous trees with additional constraints.

3.1 Non-ambiguous trees inside a fixed rectangle

In this section, we make extensive use of Remark 1 and we view non-ambiguous trees as tree-like tableaux of rectangular shape. For brevity, we write TLT for tree-like tableau in the sequel. Of particular importance for our purpose is the insertion procedure for TLTs, which gives to these objects a recursive structure. Since it is not the central purpose of the present paper, we shall not recall here all the definitions and properties of TLTs, and we refer the reader to [ABN11]. Figure 5 shows an example of a TLT of size 7. In full generality, the size of TLT is given by the numbers of dotted cells.

Moreover, in this section, we use a different drawing convention for non-ambiguous trees, inherited from the drawing convention for TLTs: we will draw non-ambiguous trees with the x-axis vertical, and the y-axis horizontal.

Given a non-ambiguous tree, its x-size (resp. y-size) may be defined as the maximum of the x-coordinate (resp. y-coordinate) of its points. The aim of this subsection is to give a formula for the number $A(k,\ell)$ of non-ambiguous trees with x-size equal to ℓ and y-size equal to k. Because the size of a TLT is given by its semi-perimeter-1, remark that the size n of such a non-ambiguous tree is given by $n = k + \ell - 1$.

We denote by $c(n,j)$ the unsigned Stirling numbers of the first kind, *i.e.* the number of permutations of size n with exactly j disjoint cycles.

We shall prove that for every integers n, ℓ, one has:

$$\sum_{k=1}^{n} c(n,k)\, A(k,\ell) = n^{\ell-1}\, n!\,. \tag{1}$$

Inverting Equation (1), we obtain that it is equivalent to Equation (2) below:

$$A(k,\ell) = \sum_{i=1}^{k} (-1)^{k-i}\, S(k,i)\, i!\, i^{\ell-1}, \tag{2}$$

where $S(k,i)$ denotes the Stirling numbers of the second kind, *i.e.* the number of partitions of a set of k elements into i non-empty parts. It is known [SW07, ABN13] that $A(k,\ell)$ is equal to the number of permutations of size $k + \ell - 1$ with exactly k strict excedances in position $1, 2, \ldots, k$. Consequently, Corollary 6.6 in [ES00] provides a proof of Equation (1). More precisely, in [ES00], Ehrenborg and Steingrímsson prove (2) by an inclusion-exclusion argument, and then deduce (1) by inversion.

Our goal is to prove Equation (1) directly and combinatorially. For this purpose, we provide combinatorial interpretations of the numbers $c(n,k)$ and $A(k,\ell)$ that appear in Equation (1) with TLTs and non-ambiguous trees respectively. As mentioned in Remark 1, non-ambiguous trees are nothing but TLTs with a rectangular shape, so that our interpretation of $c(n,k)$ and $A(k,\ell)$ is then with unified objects. This is the key in proving Equation (1) by a combinatorial approach.

Recall that by definition $A(k,\ell)$ is the number of non-ambiguous trees with x-size equal to ℓ and y-size equal to k. On the other hand, we claim that $c(n,j)$ counts the number of TLTs of size n with exactly j points in their first row. Indeed, this follows by definition of $c(n,j)$ and the fact that TLTs of size n with exactly j points in their first row are in bijection with permutations of size n with exactly j disjoint cycles. This is a consequence of Theorem 4.2 in [Bur07], re-formulated in terms of TLTs as in [ABN13].

Proposition 2. *For every integers n, ℓ, we have the following identity:*

$$\sum_{k=1}^{n} c(n,k)\, A(k,\ell) = n^{\ell-1}\, n! \tag{3}$$

where $c(n,k)$ denotes the number of TLTs of size n with exactly k points in their first row.

The proof consists in two steps:

- introduce a set $\mathcal{T}_{n,\ell}$ whose cardinality will be proved to be $n^{\ell-1}\, n!$ (Lemma 3);

- prove that the elements of $\mathcal{T}_{n,\ell}$ are in bijection with pairs (b,a) enumerated by the left-hand side of (3) (Lemma 4).

We define $\mathcal{T}_{n,\ell}$ as the set of TLTs of size $n + \ell - 1$ whose first ℓ rows are of equal length. Figure 6 shows an element of $\mathcal{T}_{5,2}$.
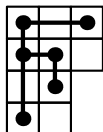

Figure 6: A tree-like tableau of $T_{5,2}$

To compute the cardinality of $\mathcal{T}_{n,\ell}$, we will use a slight variation of the insertion procedure defined in [ABN11]. This new insertion procedure depends on ℓ, and we shall call it the ℓ-*insertion procedure*.

The ℓ-insertion procedure.   Let T be an element of $\mathcal{T}_{n,\ell}$. The ℓ-special box of T is defined as

- the right-most box of the ℓ-th row, if the $(\ell+1)$-st row is strictly smaller;

- the right-most dotted box at the bottom of a column, otherwise.

Given an integer m in $\{1, \ldots, n\}$, we associate to it an edge e_m of the South-East border of T by labeling these edges from South-West to North-East. Note that we exclude the (vertical) edges $e_{n+1}, \ldots, e_{n+\ell}$ at the right of the first ℓ rows of T. As in the insertion procedure of [ABN11], if e_m is a horizontal (resp. vertical) edge of the border of T, we add a row (resp. column) to T below (resp. to the right of) e_m, composed of empty boxes except for one dotted box: the one below (resp. to the right of) e_m. Next, if the dotted box added is to the left of the ℓ-special box of T, we add a *ribbon* (a connected set of empty boxes without any 2×2 square) adjacent simultaneously to the ℓ-special box of T and to the dotted box added.

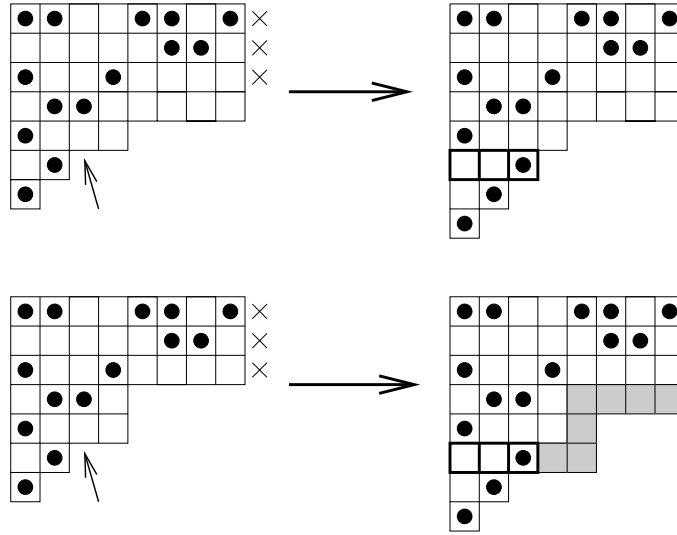Figure 7 illustrates the ℓ-insertion procedure.

Figure 7: The ℓ-insertion procedure on two examples for $\ell = 3$ and $m = 5$.

Lemma 3. *The cardinality of $\mathcal{T}_{n,\ell}$ is given by*

$$\#\mathcal{T}_{n,\ell} = n^{\ell-1}\, n!\,. \tag{4}$$

Proof. We prove the above by induction on ℓ. By [ABN11, Theorem 2.2], the number of unrestricted TLTs of size n is equal to $n!$, thus (4) is true for $\ell = 1$.

For the inductive step, we claim that the ℓ-insertion procedure gives a bijection between $\mathcal{T}_{n,\ell} \times \{1, \ldots, n\}$ and $\mathcal{T}_{n,\ell+1}$. To prove this fact, we first notice that the result of the ℓ-insertion procedure on $T \in \mathcal{T}_{n,\ell}$ and e_m with $m \in \{1, \ldots, n\}$ is a TLT whose first $\ell+1$ rows are of equal length, thus an element of $\mathcal{T}_{n,\ell+1}$. Next, as explained in [ABN11], we observe that the new box added by ℓ-insertion in an element of $\mathcal{T}_{n,\ell+1}$ is easy to recognize: it is the rightmost among dotted boxes at the bottom of a column. Thus we are able to invert the ℓ-insertion procedure. This ends the proof of the lemma. □

Now we will send bijectively an element of $\mathcal{T}_{n,\ell}$ on pairs of objects (b, a) enumerated by the left-hand side of (3). The bijection relies on the ℓ-*cut procedure* described below, and illustrated in Figure 8.

The ℓ-cut procedure. Let T be an element of $\mathcal{T}_{n,\ell}$. We first cut T by putting the first ℓ rows in a' and the next rows in b'. Now we see b' as part of a TLT. We add to it a first row whose length equals the width of a'. We observe that there is exactly one way to put dots in this row to obtain a TLT, that we denote b: we are forced to put dots in the boxes corresponding to non-empty columns in a'. Next we remove empty columns in a' to get a non-ambiguous tree with ℓ rows, denoted a.
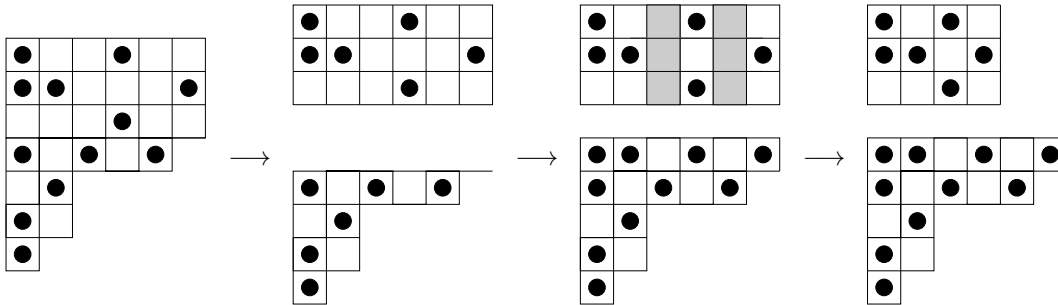

Figure 8: The ℓ-cut procedure for $\ell = 3$.

Lemma 4. *Elements of $\mathcal{T}_{n,\ell}$ are in bijection with pairs (b, a) such that*

1. *b is a TLT of size n,*

2. *a is a non-ambiguous tree with ℓ rows,*

3. *the number of dotted boxes in the first row of b and the width of a are equal.*

Proof. It should be clear the ℓ-cut procedure is invertible, which implies the lemma. □

Putting together Lemmas 3 and 4 gives a bijective proof of Proposition 2.

3.2 Non-ambiguous trees with a fixed underlying tree: a new hook formula

Let T be a binary tree. We define $NA(T)$ as the number of non-ambiguous trees A such that their underlying binary tree $\mathsf{T}(A)$ is T. The aim of this section is to get a formula for $NA(T)$: this will be done by Proposition 8, which shows that $NA(T)$ may be expressed by a new and elegant hook formula on the edges of T. To do this, we encode any non-ambiguous tree A by a triple $\Phi(A) = (T, \alpha_L, \alpha_R)$ where T is a binary tree, and α_L (resp. α_R) is a word called the left (resp. right) code of A. To distinguish the vertices of A, we label them by integers from 1 to the size of A, as shown on Figure 9.
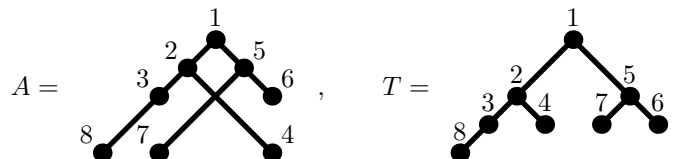

Figure 9: A non-ambiguous tree A with labeled vertices, and the associated binary tree T

6

The first entry in $\Phi(A)$ is the underlying binary tree T associated to A. Observe that we keep the labels on vertices when we extract the underlying binary tree. Now we denote by V_L (resp. V_R) the set of the end points of the left (resp. right) edges of A, which gives $V_L = \{2,3,7,8\}$ and $V_R = \{4,5,6\}$ on the example in Figure 9. The definition of non-ambiguous trees ensures that the set $\{X(v), v \in V_L\}$ is the interval $\{1, \ldots, |V_L|\}$. Thus for $i = 1, \ldots, |V_L|$, we may set $\alpha_L(i)$ as the unique label $v \in V_L$ such that $X(v) = i$, and we proceed symmetrically for α_R. On the example of Figure 9, we have: $\alpha_L = 2378$ and $\alpha_R = 564$. Our starting point is the following lemma.

Lemma 5. *The application Φ which sends A to the triple (T, α_L, α_R) is injective.*

Proof. Consider a non-ambiguous tree A with n vertices and $\Phi(A) = (T, \alpha_L, \alpha_R)$. From the definition of non-ambiguous tree, we know that the X and Y-coordinates of the root r are $X(r) = Y(r) = 0$. Moreover, for every left edge (s, t) of T we have

$$Y(s) = Y(t) \text{ and } \alpha_L(X(t)) = t,$$

namely, $X(t)$ is the position where t appears in α_L. Similarly, for every right edge (s, t), we have

$$X(s) = X(t) \text{ and } \alpha_R(Y(t)) = t,$$

namely, $Y(t)$ is the position where t appears in α_R. It is easy to check that we have $2n$ independent equations in $2n$ variables (the X and Y-coordinates of the vertices), so we get a unique non-ambiguous tree. $\square$

Lemma 5 allows us to encode a non-ambiguous tree A by a triple (T, α_L, α_R), where T is a binary tree, and α_L (resp. α_R) is a word in which every label $v \in V_L$ (resp. V_R) appears exactly once. Of course, Φ is not surjective on such triples. If we take $T = $, it should be clear that α_L is forced to be 23. Consequently, our next task is to characterize the pairs of codes (α_L, α_R) which are compatible with a given binary tree T, *i.e.* such that (T, α_L, α_R) is in the image of Φ. In order to describe this characterization, we need to define partial orders on the sets V_L and V_R. The pairs (α_L, α_R) of compatible codes will be seen to correspond to pairs of linear extensions of the posets V_L and V_R. The posets are defined as follows: given $a, b \in V_L$ (resp. V_R), we say that $a \leq b$ if and only if there exists a path in the oriented tree starting from a and ending at b. Figure 10 (with minima at the top) illustrates this notion.
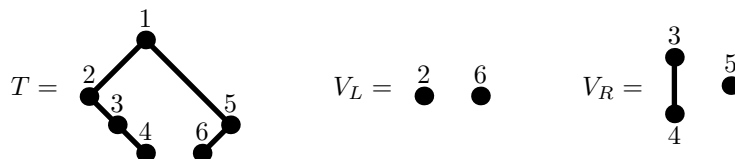
Figure 10: The posets V_L and V_R of a tree T

The next lemma is the crucial step to prove Proposition 8.

Lemma 6. *Given a binary tree T, the pairs of codes compatible with T are exactly the pairs (α_L, α_R) where α_L is a linear extension of V_L and α_R is a linear extension of V_R.*
Moreover, such pairs (α_L, α_R) are in bijection with non-ambiguous trees with underlying tree T.

Figure 11 gives these compatible codes, together with the corresponding non-ambiguous trees, in the case of the tree T of Figure 10.

Figure 11: Non-ambiguous trees of the tree T of Figure 10

Proof. Given a tree T, consider the map Φ_T defined on the set of non-ambiguous trees with underlying tree T by $\Phi_T(A) := (\alpha_L, \alpha_R)$, where $\Phi(A) = (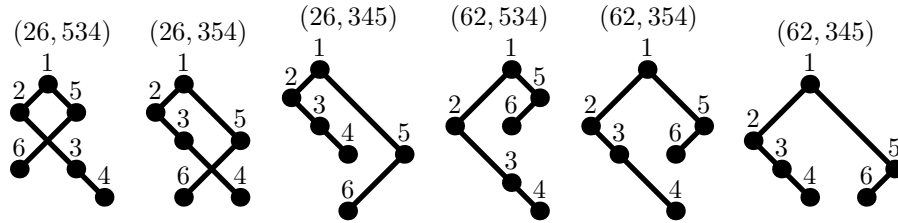T, \alpha_L, \alpha_R)$. We prove that the image of Φ_T is $\mathcal{L}(V_L) \times \mathcal{L}(V_R)$, where we denote by $\mathcal{L}(P)$ the set of linear extensions of a poset P. The second statement in Lemma 6 will then follow, since we deduce from the injectivity of Φ (Lemma 5) that Φ_T is also injective.

First, we prove that $Im\, \Phi_T \subseteq \mathcal{L}(V_L) \times \mathcal{L}(V_R)$. Without loss of generality, we will prove that $\alpha_L \in \mathcal{L}(V_L)$. We need to prove that, if $s <_{V_L} t$, then s precedes t in α_L, which we shall write $s <_{\alpha_L} t$. If $s <_{V_L} t$, there exists a path in T starting from s and ending at t. When we go through the path, the X-coordinates of the vertices remain unchanged along right edges, while they increase along left edges. Since $s \neq t$, we have $X(s) < X(t)$, which is equivalent to $s <_{\alpha_L} t$.

Now the hard part is to prove that $\mathcal{L}(V_L) \times \mathcal{L}(V_R) \subseteq Im\, \Phi_T$. Let $(\alpha_L, \alpha_R) \in \mathcal{L}(V_L) \times \mathcal{L}(V_R)$. From the triple (T, α_L, α_R), we may build a set of points in the grid, denoted A, as follows: we place a point (the root) at position $(0, 0)$ and one point for every vertex v in T at coordinates

$$\begin{cases} X(v) = i \text{ with } \alpha_L(i) = v \quad \text{and} \quad Y(v) = Y(\text{parent}(v)) & \text{if } v \in V_L; \\ X(v) = X(\text{parent}(v)) \quad \text{and} \quad Y(v) = j \text{ with } \alpha_R(j) = v & \text{if } v \in V_R. \end{cases}$$

We now prove that A is a non-ambiguous tree, whose underlying binary tree is of course T. This will follow from the three following statements, that we prove below.

  i. for every left (resp. right) edge (s, t) of T, we have $X(s) < X(t)$ (resp. $Y(s) < Y(t)$) in A;

  ii. A avoids the pattern ;

  iii. two different vertices in T correspond to points at different positions in A.

Proof of i. Without loss of generality, we show this property for the set of left edges. First of all, we define the *oldest left (resp. right) ancestor* of a vertex u to be the vertex v such that the path going from v to u contains only right (resp. left) edges and this path is the longest with this property.

For example, in Figure 12 the vertices named v are the oldest left ancestors of those named u.

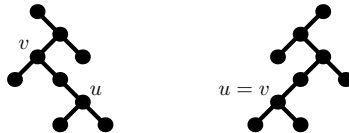Figure 12: The oldest left ancestors

Let v be the oldest left ancestor of s in T. Our construction of the set A implies that $X(v) = X(s)$. Moreover, since (s, t) is a left edge, we have $\alpha_L(X(t)) = t$. Keeping in mind these two facts, we consider the following two cases:

  • if v is the root of the tree, then $X(v) = 0$. Since $X(t) \in [1, n]$, we have $0 = X(s) < X(t)$;

8

- otherwise, v is the ending point of a left edge in T, and $\alpha_L(X(v)) = v$. Since in T there exists a path form v to t, then $v <_{V_L} t$. Moreover, we know that $\alpha_L \in \mathcal{L}(V_L)$, so $v <_{\alpha_L} t$ and $X(v) < X(t)$, implying that $X(s) < X(t)$.

Proof of ii. We proceed by way of contradiction. Suppose that there are three points $s, t, u \in A$ such that $s - t - u$ form the pattern $\overset{s \quad u}{\underset{t}{\vee}}$. We remark that these three points are different from the root, because for each of them either the X or the Y-coordinate is nonzero. Without loss of generality, we suppose that t is the end of a left edge. Let v be the oldest left ancestor of s. By construction, $Y(v) \le Y(s) < Y(t)$, and hence $t \ne v$. Since $X(v) = X(s) \ne 0$, v is the end of a left edge. The points v and t therefore both belong to V_L and have the same X-coordinate, enventhough they are distinct. This is impossible, since α_L is a linear extension of V_L, and hence all points in V_L must have different X-coordinates.

Proof of iii. We proceed by way of contradiction. Suppose that there are two different vertices $s, t \in T$ whose positions in A coincide. By construction, they cannot be the root of the tree. We have two cases:

- one of the vertices is the end of a left edge, and the other is the end of a right edge. In this case, we find an occurrence of the forbidden pattern $\vee$;

- s and t are both the end of a left edge (the other case is similar). In this case, we have $s, t \in V_L$ and $X(s) = X(t)$ which is impossible. $\qquad\square$

Now we come to the final step toward proving Proposition 8.

Lemma 7. *The Hasse diagrams of V_L and V_R are forests.*

Proof. We prove this proposition by way of contradiction. Suppose that there is a cycle in the Hasse diagram of V_R (the case of V_L is analogous). We can deduce from the poset structure that there are two paths in V_R starting from an element v and ending at w. This would imply that in the tree there are two different paths from v to w, and hence there would be a cycle in the tree. $\qquad\square$

Figure 13 shows an example of the forests obtained by computing the Hasse diagrams of V_L and V_R.

Figure 13: The Hasse diagrams $H(V_L)$ and $H(V_R)$ of V_L and V_R are forests

Proposition 8. *The number of non-ambiguous trees with underlying tree T is given by*

$$NA(T) = \frac{\#\{left\ edges\}!\ \#\{right\ edges\}!}{\displaystyle\prod_{e \in V_L} n_e \prod_{e \in V_R} n_e} \qquad (5)$$

where, for every left edge (resp. right edge) e, n_e is the number of left edges (resp. right edges) contained in the subtree whose root is the ending point of e, plus 1.

Proof. Recall that Knuth's hook formula [Knu98] gives the number of linear extensions of a poset V whose Hasse diagram is a forest: namely, the number of these linear extensions is $|V|!$ divided by the product of the hook lengths of all vertices in V. Therefore, by Lemmas 6 and 7, the number of non-ambiguous trees with underlying tree T is given by the product of the results of Knuth's hook formula applied to the Hasse diagrams of V_L and V_R. Specifically, when applying Knuth's formula to V_L (resp. V_R), the hook length of any vertex v in V_L (resp. V_R) is the number of descendants of v in V_L (resp. V_R) including v, which corresponds exactly to n_e for the left (resp. right) edge e whose end point is v. $\square$

This new hook formula is illustrated by Figure 14.

$$NA \begin{pmatrix} \cdots \end{pmatrix} = \frac{11! \cdot 6!}{1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 3 \cdot 6 \cdot 8 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 4}$$

Figure 14: A hook formula for non-ambiguous trees on an example

4 Complete non-ambiguous trees and combinatorial identities

A non-ambiguous tree is *complete* whenever its vertices have either 0 or 2 children. An example of complete non-ambiguous tree can be found in Figure 15. A complete non-ambiguous tree always has an odd number of vertices. Moreover, as in complete binary trees, a complete non-ambiguous tree with $2k+1$ vertices has exactly k internal vertices, $k+1$ leaves, k right edges and k left edges. Denote by b_k the number of complete non-ambiguous trees with k internal vertices. The sequence $(b_k)_{k \geq 0}$ is known in [Slo] as A002190 $= [1, 1, 4, 33, 456, 9460, \dots]$. We give in this section the first combinatorial interpretation for this integer sequence, in terms of complete non-ambiguous trees. Moreover, we use this interpretation to give in Propositions 9 and 11 combinatorial proofs of two identities due to Carlitz [Car63].

4.1 Enumeration of complete non-ambiguous trees, and connection to Bessel function

Denote by C_n the number of complete binary trees with n internal vertices. It is well-known that $C_n = \frac{1}{n+1}\binom{2n}{n}$ is the n-th Catalan number, and that, for every $n \geq 0$, we have the identity:

$$C_{n+1} = \sum_{i+j=n} C_i C_j. \tag{6}$$

Proposition 9 gives a variant of this identity for complete non-ambiguous trees:

Proposition 9. *For every $n \geq 0$, we have:*

$$b_{n+1} = \sum_{i+j=n} \binom{n+1}{i}\binom{n+1}{j} b_i \, b_j. \tag{7}$$

Proof. The proof of this proposition is similar to the classical proof of (6): the left (resp. right) subtree A_L (resp. A_R) of a complete non-ambiguous tree A with $n+1$ internal vertices is a complete non-ambiguous tree with i (resp. j) internal vertices, where $i+j=n$.

Figure 15 shows an example of left and right subtree of a complete non-ambiguous tree.
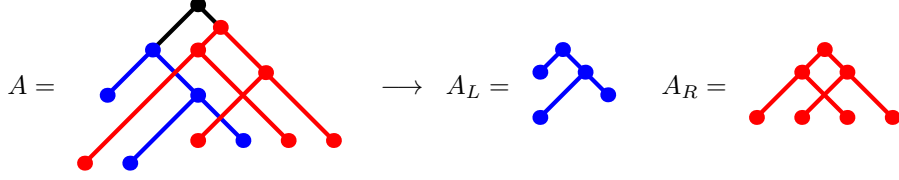

Figure 15: The root suppression in a complete non-ambiguous tree

Hence, in order to construct an arbitrary complete non-ambiguous tree A with $n+1$ internal vertices, we need to choose:

- the number i of internal vertices contained in A_L (i may range between 0 and n, the number j is equal to $n-i$);

- the complete non-ambiguous tree structure of A_L (resp. A_R) – we have b_i (resp. b_j) choices;

- the way of interlacing the right (resp. left) edges of A_L and A_R.

We denote by $u_1, u_2, \ldots, u_i$ (resp. $v_1, v_2, \ldots, v_j$) the end points of the right edges in A_L (resp. A_R) such that if $k < l$, then $Y(u_k) < Y(u_l)$ (resp. $Y(v_k) < Y(v_l)$), and by u_0 and v_0 the roots of A_L and A_R. Now, if we want to interlace the right edges in A_L with those in A_R, we need to decide at what positions we want to insert the vertices $u_1, u_2, \ldots, u_i$ with respect to $v_0, v_1, v_2, \ldots, v_j$, saving the relative order among $u_0, u_1, u_2, \ldots, u_i$ and $v_0, v_1, v_2, \ldots, v_j$. A vertex u_k can be placed either to the left of v_0, or between v_t and v_{t+1} ($0 \le t \le j-1$), or to the right of v_j.

Hence, we must choose the i positions of $u_1, u_2, \ldots, u_i$ (multiple choices of the same position are allowed) among $j+2$ possible ones. This shows that there are $\left(\binom{j+2}{i}\right) = \binom{i+j+1}{i} = \binom{n+1}{i}$ ways of interlacing the right edges of the subtrees A_L and A_R, where $\left(\binom{a}{b}\right)$ denotes the number of way of choosing b objects within a, with possible repetitions.

Analogous arguments apply to left edges. In this case, we have $\left(\binom{i+2}{j}\right) = \binom{n+1}{j}$ different interlacements. This ends the proof. $\qquad\square$

Corollary 10. *The sequence b_k satisfies the following identity*

$$\sum_{k \ge 0} b_k \frac{x^{2(k+1)}}{((k+1)!2^{k+1})^2} = -\ln\left(J_0(x)\right). \tag{8}$$

Proof. It is well known (see, e.g., [AS64]) that the Bessel function $J_0(x) = \sum_{k \ge 0} j_k x^k$ satisfies the differential equation

$$\frac{d^2 y}{dx^2} + \frac{1}{x}\frac{d\,y}{dx} + y = 0, \tag{9}$$

The first coefficients in its series expansion are $j_0 = 1$ and $j_1 = 0$.

Consider now the function $B(x) = exp\left(-\sum_{k \ge 0} b_k \frac{x^{2(k+1)}}{((k+1)!2^{k+1})^2}\right) = \sum_{k \ge 0} \beta_k x^k$. Equation (7) ensures that $B(x)$ satisfies Equation (9), *i.e.* the same second order differential equation as $J_0(x)$.

Setting $x = 0$, we have $\beta_0 = B(0) = 1 = j_0$. Moreover, in $Z(x) = -\sum_{k \ge 0} b_k \frac{x^{2(k+1)}}{((k+1)!2^{k+1})^2}$ only the even powers of x have non-zero coefficients. Hence, since $B(x) = exp\left(Z(x)\right) = \sum_{k \ge 0} \frac{Z(x)^k}{k!}$, we have $\beta_{2i+1} = 0$ for every $i \ge 0$. In particular, $\beta_1 = 0 = j_1$. These arguments imply that $B(x) = J_0(x)$. $\qquad\square$

11

4.2 Combinatorial identities

Corollary 10 shows that non-ambiguous trees provide a combinatorial interpretation – and to our knowledge, the first one – of sequence A002190 [Slo].

In [Car63], the author shows analytically that identities (7) and (10) below are equivalent. We give a combinatorial proof of this fact.

Proposition 11. *For every $n \geq 1$, we have:*

$$\sum_{k=0}^{n-1}(-1)^k \binom{n}{k+1}\binom{n-1}{k}b_k = 1. \tag{10}$$

Proof. We fix an integer n and we take $0 \leq k \leq n-1$. We define a gridded tree of size (k, n) to be a set of $2k + 1$ points placed in a $n \times n$ grid, such that Condition 2 defining non-ambiguous trees is satisfied (which means we consider a non-ambiguous tree of size $2k + 1$ embedded in a $n \times n$ grid) and such that the underlying tree is complete and that its root belongs to the first column. This implies that there are $n - k - 1$ empty columns and $n - k - 1$ empty rows, and that the first column is not empty. Figure 16 shows an example of a gridded tree of size $(2, 6)$.
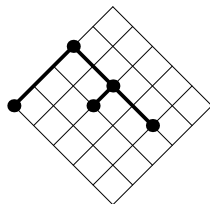
Figure 16: An example of gridded tree with 2 internal vertices drawn on a 6×6 grid

It is easy to verify that there are $\binom{n}{k+1}\binom{n-1}{k}b_k$ gridded trees of size (k, n). We call trivial gridded tree the tree of size $(0, n)$ consisting of a single vertex in $(0, 0)$. Now, for every integer n, we define an involution on the set of non trivial gridded trees. This involution associates a gridded tree of size (k, n) with a gridded tree either of size $(k - 1, n)$ or $(k + 1, n)$.

To define this involution, consider a gridded tree of size (k, n) and add a virtual root at position $(-1, 0)$; the previous root becomes the left child of the virtual root. Now consider the path starting from the virtual root, going down through the tree, turning at each internal vertex, and ending at a leaf. This path is unique. There are two cases:

1. the path does not cross an empty row, nor an empty column: we erase the leaf and its parent from the tree, getting a new gridded tree of size $(k - 1, n)$. We can always erase the leaf and its parent, except if the parent is the virtual root. This happens only if the tree is the trivial gridded tree. As we restricted to non trivial gridded trees, this case never happens.

2. the path crosses an empty row or an empty column: we choose the first empty row or column met while visiting the path. Without loss of generality, we suppose that it is a column, say c. Then, we add a new vertex v at the position where c crosses the path, and we add in the same column a new leaf (whose parent is v) in the topmost empty row. While visiting the path, we did not meet an empty row. Since there are as many empty rows as empty columns, there is always an empty row below v. This operation gives rise to a new gridded tree of size $(k + 1, n)$.

Remark that adding (resp. removing) a leaf and its parent p in (resp. from) a gridded tree following the previous algorithm does not remove (resp. add) any empty row or column that crosses the path from the virtual root to p. For this reason, this operation is an involution. Figure 17 shows how the involution acts on two examples. □
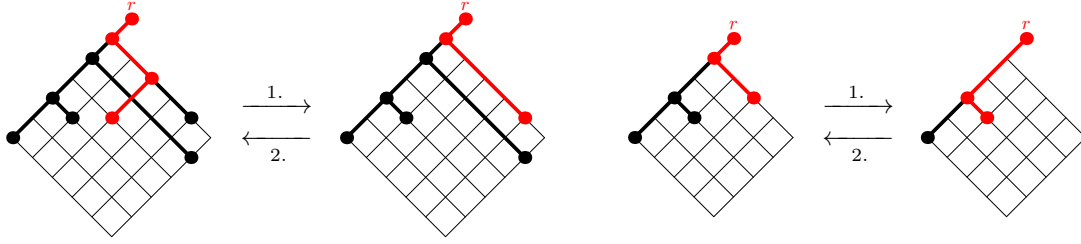
Figure 17: The involution acting on two examples of non trivial gridded trees

We now state and prove an identity satisfied by Catalan numbers, whose proof is similar to Proposition 11, and which will be used to get Corollary 13.

Proposition 12. *For every $n \geq 1$, we have:*

$$\sum_{k=0}^{n}(-1)^{n+k}\binom{n+k}{n-k}C_k = 0. \tag{11}$$

Proof. We define a (k, ℓ)-*labeled binary tree* as a complete binary tree with k internal vertices together with a labeling of its $2k + 1$ (internal or external) vertices by non-negative integers, such that the sum of these labels is equal to ℓ. Since $\binom{n+k}{n-k}$ is the number of ways to choose $n - k$ elements with repetition in a set of cardinality $2k + 1$, we may interpret $\binom{n+k}{n-k}C_k$ combinatorially as the number of $(k, n - k)$-labeled binary trees. Indeed, a $(k, n - k)$-labeled binary tree may be described by a complete binary tree with k internal vertices, together with a choice of $n - k$ vertices (with possible repetition) among its $2k + 1$ (internal or external) vertices, any vertex receiving label i when it has been chosen i times.

Now the proof is based on an involution defined on $(k, n-k)$-labeled binary trees, for $0 \leq k \leq n$, which modifies the parameter k by ± 1. This involution is defined in a way similar to the one used in the proof of Proposition 11. Let us consider a $(k, n - k)$-labeled binary tree. We add to it a virtual root and consider the path starting from the virtual root, going down through the tree, turning at each internal vertex, and ending at a leaf. Two cases occur:

1. If all the vertices on this path are labeled by zero, then we erase the leaf y at the end of this path, together with its parent, and we increase by one the label of the sibling of y. This produces a $(k - 1, n - k + 1)$-labeled binary tree.

2. Otherwise we consider the first time this path encounters a vertex v that has a non-zero label. We decrease by one the label of v and we add a new vertex x on the edge between v and its parent, having two children: v and a new leaf y (the new sibling of v). The labels associated to the new vertices x and y are zero, so that the tree obtained is a $(k + 1, n - k - 1)$-labeled binary tree.

Figure 18 illustrates this operation on two examples (labels zero are omitted on this figure). It is clear that we define in this way an involution that modifies the parameter k by ± 1, which implies (11). □
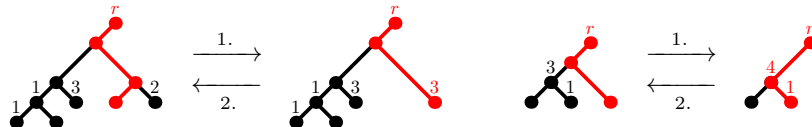

Figure 18: The involution acting on two examples of $(4, 7)$-labeled trees

We are now able to obtain a new identity involving Catalan numbers and the sequence b_k, which has been conjectured by P. Hanna (see [Slo], sequence A002190).

Corollary 13. *For every $n \geq 1$, we have:*

$$\sum_{k=0}^{n} (-1)^k b_k C_k \binom{n+k}{n-k}^2 = 0. \tag{12}$$

Proof. We fix an integer $n_0 \geq 1$ and define the matrix $D(n_0) = (D(n_0)_{i,j})_{0 \leq i,j \leq n_0}$ as follows:

$$D(n_0)_{i,j} = \begin{cases} 0 & \text{if } i = 0 \\ (-1)^j \binom{i+j}{i-j}^2 C_j & \text{otherwise} \end{cases}.$$

We also set $\underline{\mathbf{b}}(n_0)$ to be the column vector whose entries are $(b_0, b_1, \ldots, b_{n_0})$. With this notations, Identity (12) holds for every $n = 1, 2 \ldots, n_0$ if and only if $D(n_0) \cdot \underline{\mathbf{b}}(n_0) = \underline{\mathbf{0}}(n_0)$, where $\underline{\mathbf{0}}(n_0)$ is the column vector whose $n_0 + 1$ entries are all 0. Analogously, we define the matrix $A(n_0) = \left((-1)^j \binom{i+1}{j+1} \binom{i}{j} \right)_{0 \leq i,j \leq n_0}$ and set $\underline{\mathbf{1}}(n_0)$ to be the column vector whose $n_0 + 1$ entries are all 1. Then, Identity (10) for all n from 1 to $n_0 + 1$ yields that $A(n_0) \cdot \underline{\mathbf{b}}(n_0) = \underline{\mathbf{1}}(n_0)$. Because the matrix $A(n_0)$ is lower triangular with ± 1 entries on the diagonal, it is invertible, and we can write $\underline{\mathbf{b}}(n_0) = A(n_0)^{-1} \cdot \underline{\mathbf{1}}(n_0)$. Hence, setting $H(n_0) = (H(n_0)_{i,j})_{0 \leq i,j \leq n_0} = D(n_0) \cdot A(n_0)^{-1}$, to conclude the proof of Corollary 13, it is enough to prove that:

$$H(n_0) \cdot \underline{\mathbf{1}}(n_0) = \underline{\mathbf{0}}(n_0). \tag{13}$$

Notice first that $H(n_0)_{0,j} = 0$ for any $n_0 \geq 1$ and $0 \leq j \leq n_0$. This follows immediately from the definition of $H(n_0) = D(n_0) \cdot A(n_0)^{-1}$ and the fact that $D(n_0)_{0,j} = 0$ for all $0 \leq j \leq n_0$. The main part of the proof is now to show that for $1 \leq i \leq n_0, 0 \leq j \leq n_0$, we have $H(n_0)_{i,j} = (-1)^{i+j} \binom{i+j}{i-j} C_j$. Indeed, Proposition 12 then ensures that

$$\text{for all } i \geq 1, \quad \sum_{j=0}^{i} (-1)^{i+j} \binom{i+j}{i-j} C_j = 0, \quad \text{hence, for all } 1 \leq i \leq n_0, \quad \sum_{j=0}^{i} H(n_0)_{i,j} = 0.$$

Notice that for $j > i$, $\binom{i+j}{i-j} = 0$. Consequently, the above equality is equivalent to

$$\text{for all } 1 \leq i \leq n_0, \quad \sum_{j=0}^{n_0} H(n_0)_{i,j} = 0.$$

Combined with $H(n_0)_{0,j} = 0$ for all $0 \leq j \leq n_0$, this yields $H(n_0) \cdot \underline{\mathbf{1}}(n_0) = \underline{\mathbf{0}}(n_0)$ as desired.

So let us focus on proving that for $1 \leq i \leq n_0, 0 \leq j \leq n_0$, $H(n_0)_{i,j} = (-1)^{i+j} \binom{i+j}{i-j} C_j$.

Since $H(n_0)$ is characterized by $H(n_0) \cdot A(n_0) = D(n_0)$, we have to prove that, for every i, j such that $1 \leq i \leq n_0$ and $0 \leq j \leq n_0$,

$$\sum_{k=0}^{n_0} (-1)^{i+k} \binom{i+k}{i-k} \binom{k+1}{j+1} \binom{k}{j} C_k = \binom{i+j}{i-j}^2 C_j. \tag{14}$$

If $k > i$, the summand vanishes. Moreover, it is easy to check that, if $j > i$, both sides in Identity (14) vanish, since all terms are 0. Hence, to conclude the proof, it is sufficient to prove that the following identity holds for any $i > 0$ and $0 \leq j \leq i$:

$$\sum_{k=0}^{i} (-1)^{i+k} \binom{i+k}{i-k} \binom{k+1}{j+1} \binom{k}{j} C_k = \binom{i+j}{i-j}^2 C_j \tag{15}$$

For this purpose, for any $i > 0$, $k \geq 0$ and $0 \leq j \leq i$, we define

$$F_j(i,k) = \frac{(-1)^{i+k}\binom{i+k}{i-k}\binom{k+1}{j+1}\binom{k}{j}C_k}{\binom{i+j}{i-j}^2 C_j},$$

and we show that for any $i > 0$ and $0 \leq j \leq i$

$$\sum_{k=0}^{i} F_j(i,k) = 1.$$

In order to prove this identity, we apply the WZ method described in [PWZ96]. We look for a function $G_j(i,k)$ that satisfies the following identity:

$$F_j(i+1,k) - F_j(i,k) = G_j(i,k+1) - G_j(i,k). \tag{16}$$

Using the package `Gosper` of `Mathematica`, we determine that $G_j(i,k)$ should be defined by

$$G_j(i,k) = \frac{(-1)^{i+k}\binom{i+k+1}{i-k+1}\binom{k+1}{j+1}\binom{k}{j}C_k}{\binom{i+j}{i-j}^2 C_j} \cdot \frac{2(i+1)(k-j)^2}{(i+j+1)^2(i+k+1)}.$$

Indeed we then have

$$F_j(i+1,k) - F_j(i,k) - G_j(i,k+1) + G_j(i,k) = w_j(i,k) \cdot \frac{(-1)^{i+k}\binom{i+k+1}{i-k+1}\binom{k+1}{j+1}\binom{k}{j}C_k}{\binom{i+j}{i-j}^2 C_j},$$

where the term

$$w_j(i,k) = -\frac{(i-j+1)^2}{(i+j+1)^2} - \frac{i-k+1}{i+k+1} + \frac{2(i+1)(i-k+1)}{(i+j+1)^2} + \frac{2(i+1)(k-j)^2}{(i+k+1)(i+j+1)^2}$$

can be proved to be equal to zero by straightforward computations.

Now, if we sum up both sides of Identity (16) over $0 \leq k \leq i+1$, we can easily see that the right hand side telescopes to $G_j(i,i+2) - G_j(i,0) = 0$. Indeed, we have that $G_j(i,i+2) = 0$, since the binomial coefficient $\binom{i+k+1}{i-k+1}$ vanishes for $k = i + 2$. And we also have $G_j(i,0) = 0$, since $\binom{k}{j}$ vanishes for $k = 0$ and $j > 0$, and $(k-j)^2 = 0$ for $k = j = 0$. This implies that $\sum_{k=0}^{i+1}\left(F_j(i+1,k) - F_j(i,k)\right) = 0$. We remark that $F_j(i,i+1) = 0$, since $\binom{i+k}{i-k}$ vanishes if $k = i+1$, and we deduce that $f_j(i) := \sum_{k=0}^{i} F_j(i,k)$ is independent of i. Observing finally that $f_j(j) = F_j(j,j)$ (all other terms vanish, since $\binom{k}{j} = 0$ for $k < j$), we deduce that $f_j(j) = 1$, and we get the assertion. $\qquad\square$

5 A new bijection between trees and parallelogram polyominoes

We recall that a parallelogram polyomino of size n is a pair of lattice paths of length $n + 1$ with south-west and south-east steps starting at the same point, ending at the same point, and never meeting each other. Figure 19 shows some examples of parallelogram polyominoes of size 4. The two paths defining a given parallelogram polyomino delimit a connected set of boxes. We will consider the parallelogram polyomino from this point of view.
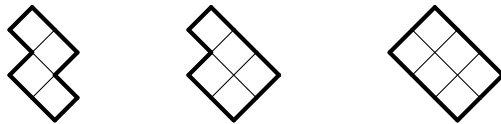
Figure 19: Example of parallelogram polyominoes of size 4

We now describe a bijection between parallelogram polyominoes of size n and binary trees with n vertices by showing that a parallelogram polyomino hides a non-ambiguous tree. In Subsection 5.1 we describe an injective map Ψ from parallelogram polyominoes to trees. In Subsection 5.2 we describe a map Λ from trees to parallelogram polyominoes, and prove in Subsection 5.3 that it is the inverse of Ψ.

5.1 The map Ψ from parallelogram polyominoes to trees

Given a parallelogram polyomino P, consider the set $S(P)$ of dots defined as follows:

- we enlighten P from north-west to south-east and from north-east to south-west;

- we put a dot in the enlightened boxes.

Lemma 14. *For any parallelogram polyomino P, $S(P)$ is a non-ambiguous tree.*

Proof. The above construction ensures that it is impossible that all three points in the pattern are enlightened. Moreover, only the northernmost box in the parallelogram polyomino can be enlightened twice. This implies that every dot (except for the one in the northernmost box) has a parent. $\qquad\square$

Let Ψ be the application that associates to any parallelogram polyomino of size n the underlying binary tree of $S(P)$ (that has n vertices). An example of this application is shown in Figure 20.
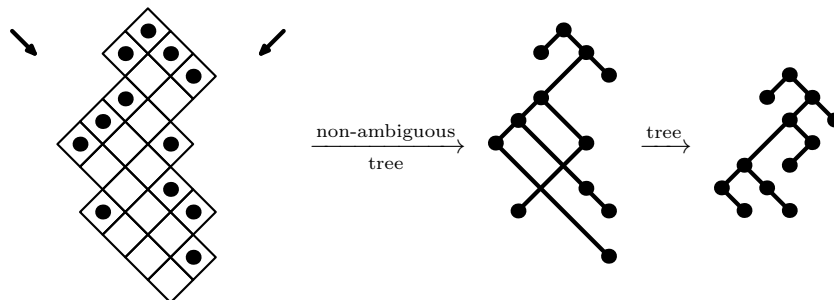
Figure 20: Parallelogram polyominoes are just a way of drawing a binary tree in the plane

Proposition 15. *The map Ψ is injective.*

Proof. Parallelogram polyominoes may be constructed incrementally, by adding, at every step of the construction process, one step to each of the two paths defining the parallelogram polyomino. Along the construction of any parallelogram polyomino P, we may build simultaneously a non-ambiguous tree whose underlying binary tree is $\Psi(P)$. More precisely, when adding one step to each of the two paths defining P, we add the enlightened dot(s) corresponding to the inserted steps, when needed. Figure 21 shows an example of this construction.
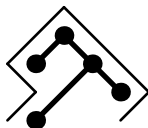
Figure 21: An example of parallelogram polyomino with its tree under construction

Considering two different parallelogram polyominoes P and P', there is a first step in the construction process when the new step added to one path in the construction of P differs from its homologous in P'. We prove that this implies that $\Psi(P) \neq \Psi(P')$. Indeed, w.l.o.g, the step added to P is SW-oriented, while the corresponding step added to P' is SE-oriented. This means that only one of these steps is associated with a new dot, connected to its parent v. The dot v exists in both trees $\Psi(P)$ and $\Psi(P')$, but it does not have the same number of children in both trees. $\qquad\square$

Since it is known that there are as many parallelogram polyominoes of size n as binary trees with n vertices, we deduce from Proposition 15 that the map Ψ is a bijection. However, it is actually possible to describe the inverse of Ψ. In what follows, we first describe a map Λ from binary trees to parallelogram polyominoes, and we next prove that it is the inverse of Ψ.

5.2 The map Λ from trees to parallelogram polyominoes

The map Λ we define here builds a parallelogram polyomino of size n from any binary tree T with n vertices. The main part of the construction is to associate to T a specific non-ambiguous tree $A(T)$ whose underlying binary tree is T. The parallelogram polyomino $\Lambda(T)$ is then immediately obtained from $A(T)$.

5.2.1 Building the non-ambiguous tree $A(T)$

The main difficulty in describing $A(T)$ is to decide where the points corresponding to the vertices of T should be drawn in the plane. Like in Subsection 3.2, their relative position is encapsulated in two total orders α_L and α_R on the set of vertices of T that are the end of a left or right edge. We will define these orders in Equation (17) p.19, but it requires some auxiliary definitions and lemmas.

Let T be a tree and T_r be the tree obtained by grafting T as the right child of a new vertex r called the virtual root. We define two labelings of the vertices of T: the turn labeling and the zigzag labeling.

The *turn labeling*, denoted by t, is the application on the vertices s of T defined by:

$$t(s) = \text{the number of turns in the path from } r \text{ to } s \text{ in } T_r.$$

For example, in Figure 22, every vertex s is labeled by $t(s)$.

Figure 22: The turn labeling

The *zigzag labeling*, denoted by z, is the application on the vertices s of T defined by:

$$z(s) = |E(T, s)|$$

where $E(T, s)$ is defined on T_r as follows:

- let $\mathcal{P}$ be the path from the virtual root r to s;

- let $s_1, s_2, \ldots, s_k$ be the vertices of $\mathcal{P}$ where $\mathcal{P}$ is turning;

- $E(T, s)$ is the set of all vertices x of T except those such that one of the followings holds:

  - $x \neq s$ is a descendant of s, or
  - x belongs to the right (resp. left) subtree of s_i and the path $\mathcal{P}$ takes a left (resp. right) turn at s_i.

Because $E(T, s)$ can be seen as obtained from T by removing some vertices with all their descendants, it is straightforward that the vertices remaining in $E(T, s)$ always form an induced subtree of T.

To illustrate the above definitions, Figure 23 shows a tree T with the induced subtree $E(T, s)$ associated to the vertex s of T, together with the complete zigzag labeling of T.
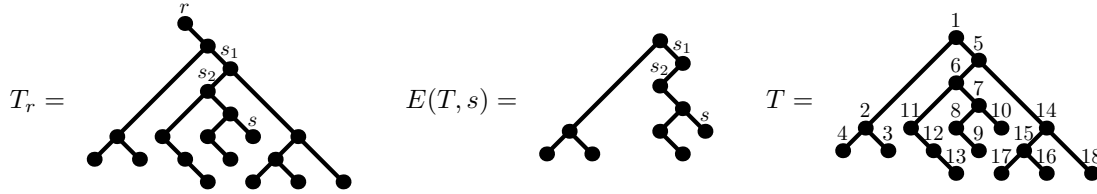
Figure 23: The zigzag labeling

With the turn and zigzag labelings of T, we will define (see Equation (17) p.19) the orders α_L and α_R on the vertices of T from which $A(T)$ (and later $\Lambda(T)$) is deduced.

An important fact in the construction is that the zigzag labeling is a bijection[1] between the set of vertices of T and $\{1, 2, \ldots, n\}$, n denoting the number of vertices of T. This will follow as a consequence of Lemma 16 below, for which we introduce some more definitions.

Let u and v be two vertices of a tree T. The *lowest common ancestor* of u and v is the last common vertex between the path going from the root to u and the path going from the root to v. We denote by $LCA(u, v)$ that vertex.

Figure 26 p.22 shows an example of lowest common ancestor.

Lemma 16. *Let u and v be two different vertices of a tree T, then*

$$E(T, u) \subsetneq E(T, v) \text{ or } E(T, v) \subsetneq E(T, u).$$

Moreover, $E(T, u) \subsetneq E(T, v)$ holds if and only if the following holds:

$$\left(\begin{array}{c} u = LCA(u, v) \\[1mm] or \\[1mm] \left[u, v \neq LCA(u, v) \quad and \quad LCA(u, v) \text{ is in } V_L \quad and \quad \begin{array}{c} u \text{ is in the right} \\ \text{subtree of } LCA(u, v) \end{array} \right] \\[1mm] or \\[1mm] \left[u, v \neq LCA(u, v) \quad and \quad LCA(u, v) \text{ is in } V_R \quad and \quad \begin{array}{c} u \text{ is in the left} \\ \text{subtree of } LCA(u, v) \end{array} \right] \end{array} \right)$$

---

[1] In fact this labeling may also be obtained by performing a kind of "zigzag" depth-first search of T.

Recall from Subsection 3.2 that V_L (resp. V_R) denotes the set of the end points of the left (resp. right) edges of T.

Proof. Let u, v be two different vertices of a tree T. By definition of $LCA(u, v)$, there exists a unique vertex $\alpha \in \{u, v\}$ (we denote by β the other vertex such that $\{u, v\} = \{\alpha, \beta\}$) such that one of the three following cases occurs:

1. $\alpha = LCA(u, v)$

2. $LCA(u, v) \notin \{u, v\}$ and $LCA(u, v) \in V_L$ and α is in the right subtree of $LCA(u, v)$

3. $LCA(u, v) \notin \{u, v\}$ and $LCA(u, v) \in V_R$ and α is in the left subtree of $LCA(u, v)$

We will now prove that cases 1, 2 and 3 imply $E(T, \alpha) \subsetneq E(T, \beta)$. For the first case, $E(T, \alpha) \subsetneq E(T, \beta)$ is a direct consequence of the definition of E. For the two other cases, we just have to remark that

- the path from the virtual root to α turns at $LCA(\alpha, \beta)$,

- the path from the virtual root to β goes through but doesn't turn at $LCA(\alpha, \beta)$,

to conclude that $E(T, \alpha) \subsetneq E(T, \beta)$. Now, Lemma 16 follows immediately. $\square$

Corollary 17. *The labeling z is a bijection between the set of vertices V of T and $\{1, 2, \ldots, |V|\}$. Moreover, for all vertices u and v in V, we have: $z(u) < z(v) \iff E(T, u) \subsetneq E(T, v)$.*

Corollary 17 is the key to defining the total orders α_L and α_R on the vertices of T, from which $A(T)$ and $\Lambda(T)$ will be derived.

We define the partial orders α_L (resp. α_R) on the ends of the left (resp. right) edges of T by the relation:

$$
\begin{array}{cc}
u <_{\alpha_L} v \\
(\text{resp. } u <_{\alpha_R} v)
\end{array}
\qquad \text{if and only if} \qquad (t(u), z(u)) <_{\text{lex}} (t(v), z(v)) \qquad (17)
$$

where $<_{\text{lex}}$ is the lexicographic order relation. Because z is a bijection, α_L and α_R are actually *total* orders.

For any binary tree T, we set $A(T)$ to be the triple (T, α_L, α_R).

Lemma 18. *For any binary tree T, $A(T)$ defines a non-ambiguous tree.*

Proof. Let us denote $A(T) = (T, \alpha_L, \alpha_R)$. We prove that α_L (resp. α_R) is a linear extension of the poset V_L (resp. V_R) of T (see Subsection 3.2 for the definition of these posets). By Lemma 6 (p. 7), this ensures that (T, α_L, α_R) encodes a non-ambiguous tree of shape T.

Let u and v be two vertices of V_L (resp. V_R) such that $u <_{V_L} v$ (resp. $u <_{V_R} v$). By definition, there exists a path from vertex u of V_L (resp. V_R) to vertex v of V_L (resp. V_R). We deduce that $t(u) \leq t(v)$. Therefore:

- if $t(u) < t(v)$ then, by definition of α_L, $u <_{\alpha_L} v$ (resp. $u <_{\alpha_R} v$);

- if $t(u) = t(v)$ then, since there is a path from u to v, we have $u = LCA(u, v)$. We deduce from Lemma 16 that $z(u) < z(v)$. Hence, $u <_{\alpha_L} v$ (resp. $u <_{\alpha_R} v$).

We conclude that α_L (resp. α_R) is a linear extension of V_L (resp. V_R). $\square$

5.2.2 From $A(T)$ to the parallelogram polyomino $\Lambda(T)$

In the sequel, by the notation [figure] (resp. [figure]) we mean that there are two cells at coordinates (x_1, y_1) and (x_2, y_2) with $x_1 > x_2$ and $y_1 < y_2$ that contain dots (or vertices), and such that the cell at coordinates (x_1, y_2) does not contain (resp. contains) a dot (or vertex). Notice that it is not necessary that $x_1 = x_2 + 1$ nor that $y_1 = y_2 - 1$ to have an occurrence of either the pattern [figure] or [figure].

For any non-ambiguous tree A, consider the operation which converts every pattern [figure] to a pattern [figure]. We shall call this operation the *filling* of A, and denote it $F(A)$.

For any binary tree T, we define $\Lambda(T) = F(A(T))$. Figure 24, shows all the steps of the construction.
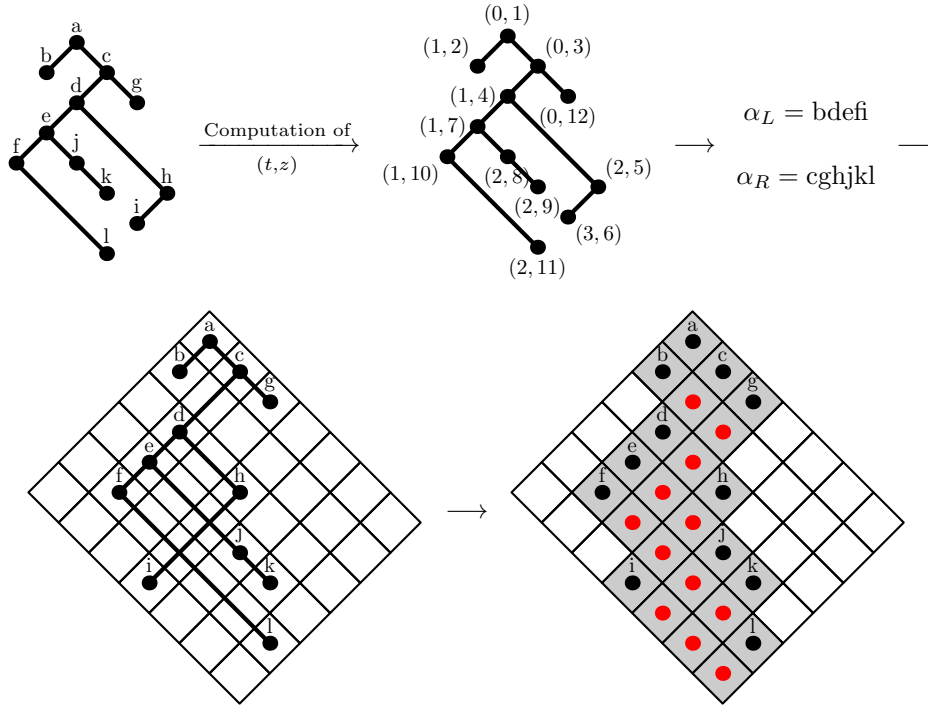
Figure 24: The map Λ

Lemma 19 and Proposition 21 below prove that for any binary tree T, the set of points $\Lambda(T)$ is indeed a parallelogram polyomino.

Lemma 19. *Let A be a non-ambiguous tree, and define V_L and V_R as usual. $F(A)$ is a parallelogram polyomino as soon as for any two different vertices u, v in V_L (resp. V_R) we have $X(u) < X(v) \Rightarrow Y(u) \leq Y(v)$ (resp. $Y(u) < Y(v) \Rightarrow X(u) \leq X(v)$).*

Proof. In the proof of this lemma, we use a characterization of parallelogram polyominoes (see [DDD95] for more details): a parallelogram polyomino is a polyomino P (*i.e.*, a finite set of unit cells on the grid $\mathbb{N} \times \mathbb{N}$ – oriented as shown on Figure 1 – that is edge-connected) such that:

- P is row and column convex, *i.e.* each of its rows and columns is connected;

- if we consider the minimum rectangle R bounding P, then P contains the top-most and bottom-most corners of R.

Suppose that for every $u, v \in V_L$ we have $X(u) < X(v) \Rightarrow Y(u) \leq Y(v)$ and that, for every $t, w \in V_R$, we have $Y(t) < Y(w) \Rightarrow X(t) \leq X(w)$. Recall that $\{X(v), v \in V_L\}$ is the interval $\{1, \ldots, |V_L|\}$. Without loss of generality, we will show that $F(A)$ is column convex. Suppose that $F(A)$ contains the two cells (a, y_0) and (b, y_0), $a < b$. We want to show that $F(A)$ contains all the cells (c, y_0), with $a < c < b$. The two cells (a, y_0) and (b, y_0) can correspond either to a vertex of A or to the 0 of a pattern ◈⟍◈⟋◇ in A. We study the case where both cells correspond to occurrences of the pattern ◈⟍◈⟋◇ , the other three possible cases being analogous. Denote by g the southwestern-most vertex of A such that $X(g) < a$ and $Y(g) = y_0$ and by h the southeastern-most vertex of A such that $X(h) = b$ and $Y(h) < y_0$. Let $h' \in V_L$ be the oldest left ancestor of h. We choose now a cell $f = (c, y_0)$, where $a < c < b$. Because $X(h') = b$, $c < b$ and $\{X(v), v \in V_L\} = \{1, \ldots, |V_L|\}$, there exists some vertex $h'' \in V_L$ such that $X(h'') = c$. But by our assumption, we deduce from $X(h'') < X(h')$ that $Y(h'') \leq Y(h') < y_0$. Hence, h'', f, and g give rise to an occurrence of the pattern ◈⟍◈⟋◇ , as shown in Figure 25.
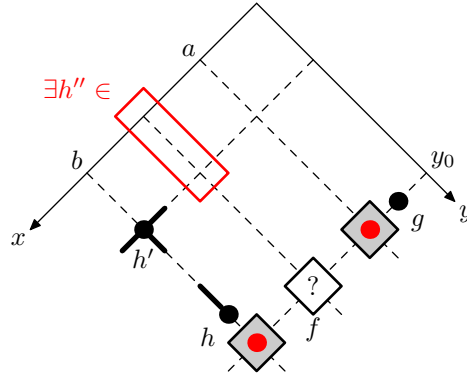

Figure 25: The positions of h'', f and g

This implies that f belongs to $F(A)$, and hence $F(A)$ is column convex.

Now we will prove that $F(A)$ is connected. Consider two cells $c_1, c_2 \in F(A)$. We suppose that both c_1 and c_2 correspond to the 0 of two patterns of type ◈⟍◈⟋◇ , the other possible cases being easier to prove. Denote by d_i, $i = 1, 2$, the southwestern-most vertex of A such that $X(d_i) < X(c_i)$ and $Y(d_i) = Y(c_i)$. Since $F(A)$ is column convex, it contains the vertical strip S_1 (resp. S_2) joining d_1 and c_1 (resp. d_2 and c_2). Now d_1 and d_2 belong to A, and they are connected by a sequence of edges, that correspond to a subset S of $F(A)$. Indeed, $F(A)$ is both row and column convex, and it contains all the cells corresponding to the vertices of A. Then, the two cells c_1 and c_2 are connected by $S_1 \cup S_2 \cup S$.

Finally, we consider the minimum rectangle R bounding $F(A)$. The root of A occupies the top-most corner, which is therefore contained in $F(A)$. Moreover, it is obvious that both the southwestern-most row and the southeastern-most column of R contain a vertex of A, and for this reason, the bottom-most corner of R is the 0 in a pattern ◈⟍◈⟋◇ of A. Therefore, $F(A)$ contains also the bottom-most corner of R.

All these arguments prove that $F(A)$ is a parallelogram polyomino. □

Remark 20. *We may prove that the condition in Lemma 19 is also necessary: if $F(A)$ is a parallelogram polyomino, then for any two different vertices u, v in V_L (resp. V_R) we have $X(u) < X(v) \Rightarrow Y(u) \leq Y(v)$ (resp. $Y(u) < Y(v) \Rightarrow X(u) \leq X(v)$).*

Proposition 21. *For any binary tree T, $\Lambda(T)$ is a parallelogram polyomino.*

One additional definition is used in our proof of Proposition 21. Let u be the end of a left (resp. right) edge of T. The *direct oldest ancestor* of a vertex u is oldest ancestor of u that can reach u using left (resp. right) edges only. We denote by $DOA(u)$ that vertex.

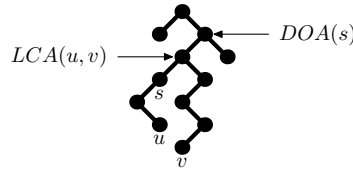Figure 26 shows an example of direct oldest ancestor.

Figure 26: Example of lowest common ancestor and direct oldest ancestor

Proof. Let T be any binary tree, and set $A(T) = (T, \alpha_L, \alpha_R)$. Because $\Lambda(T) = F(A(T))$, it is enough to prove that $A(T)$ satisfies the condition of Lemma 19. Let u, v be two different vertices of V_L such that $X(u) < X(v)$. (The case where $u, v \in V_R$ and $Y(u) < Y(v)$ is analogous.)

We want to prove that $Y(u) \leq Y(v)$. As u and v are in V_L, we have $Y(u) = Y(DOA(u))$ and $Y(v) = Y(DOA(v))$, and we shall rather prove that $Y(DOA(u)) \leq Y(DOA(v))$.

If $DOA(u) = DOA(v)$, the above claim is clear. Let us therefore assume that $DOA(u) \neq DOA(v)$.

Because u and v are in V_L, we obtain immediately that $t(DOA(u)) = t(u) - 1$ and $t(DOA(v)) = t(v) - 1$. Moreover, since $X(u) < X(v)$ Lemma 18 and the definitions of Subsection 3.2 ensure that $u <_{\alpha_L} v$. We now distinguish two cases, following the definition of α_L.

Case 1: $t(u) < t(v)$.

- If $DOA(u)$ is the root then $Y(DOA(u)) = 0$ and $Y(DOA(u)) \leq Y(DOA(v))$.

- If $DOA(v)$ is the root, since $t(u) < t(v)$, $DOA(u)$ is the root and we conclude as above.

- Otherwise, $DOA(u)$ and $DOA(v)$ are the end of a right edge. From $t(u) < t(v)$, we obtain that $t(DOA(u)) < t(DOA(v))$. Hence $DOA(u) <_{\alpha_R} DOA(v)$. As before, we deduce that $Y(DOA(u)) < Y(DOA(v))$.

Case 2: $t(u) = t(v)$ **and** $z(u) < z(v)$.
Since $t(u) = t(v)$ we have $t(DOA(u)) = t(DOA(v))$. We deduce that the following equivalence holds: $z(DOA(u)) < z(DOA(v)) \Leftrightarrow DOA(u) <_{\alpha_R} DOA(v) \Leftrightarrow Y(DOA(u)) < Y(DOA(v))$. We shall therefore prove that $z(DOA(u)) < z(DOA(v))$. This will follow from the two claims below:

1. $LCA(u, v) \neq u$ and $LCA(u, v) \neq v$.

2. $LCA(u, v) = LCA(DOA(u), DOA(v))$.

Proof of Claim 1. As u, v are in V_L, u and v can't be the root. We deduce that $DOA(u) \neq u$ and $DOA(v) \neq v$. Because we assumed that $DOA(u) \neq DOA(v)$ and $t(u) = t(v)$, we deduce that $LCA(u, v) \neq u$ and $LCA(u, v) \neq v$.

Proof of Claim 2. This is also a consequence of $DOA(u) \neq DOA(v)$ and $t(u) = t(v)$.

- If $LCA(DOA(u), DOA(v)) \neq DOA(u)$ and $LCA(DOA(u), DOA(v)) \neq DOA(v)$, we apply Lemma 16 simultaneously on u and v on one side, and on $DOA(u)$ and $DOA(v)$ on the other side. Because $LCA(u, v) = LCA(DOA(u), DOA(v))$, we deduce that: $E(T, u) \subsetneq E(T, v) \Leftrightarrow E(T, DOA(u)) \subsetneq E(T, DOA(v))$. With Corollary 17, this is equivalent to $z(u) < z(v) \Leftrightarrow z(DOA(u)) < z(DOA(v))$. As $z(u) < z(v)$, we conclude that $z(DOA(u)) < z(DOA(v))$.

- If $LCA(DOA(u), DOA(v)) = DOA(u)$, then we apply Lemma 16 on $DOA(u)$ and $DOA(v)$ to deduce that $E(T, DOA(u)) \subsetneq E(T, DOA(v))$, hence $z(DOA(u)) < z(DOA(v))$.

- We finally prove that the case $LCA(DOA(u), DOA(v)) = DOA(v)$ is not possible. Indeed, assume that $LCA(DOA(u), DOA(v)) = DOA(v)$. If $DOA(v)$ were the root of T, then we would have $z(DOA(v)) = 1$, which is a contradiction to $z(u) < z(v)$. So $DOA(v)$ is the end of a left (resp. right) edge. Hence by definition, v is in the right (resp. left) subtree of $DOA(v)$. But from Claim 2, we have $LCA(u, v) = DOA(v)$. Lemma 16 then ensures that $z(v) < z(u)$, which is a contradiction to $z(u) < z(v)$. $\qquad\square$

5.3 Proving that Λ is the inverse of Ψ

We want to prove that the maps Ψ and Λ defined in the previous subsections are inverse of one another. First of all, we define a *diagram* to be a finite collection of cells in the square grid $\mathbb{N} \times \mathbb{N}$. Then, define the set D_0 (resp. D_1) of all the diagrams that avoid the pattern

(resp.). We remark that every parallelogram polyomino belongs to D_0, since it is convex and contains the bottom-most point of the minimum rectangle bounding it. Moreover, every non-ambiguous tree belongs to D_1 by definition.

We extend the map S defined in Subsection 5.1 to the set D_0, defining $\hat{S}$ as follows: for every diagram $U \in D_0$, $\hat{S}(U)$ is the diagram obtained by replacing every 1 of U occurring as the bottom-most 1 of a pattern by a 0. Analogously, we can define the extension $\hat{F}$ of F to the set D_1: for every diagram $W \in D_1$, $\hat{F}(W)$ is the diagram obtained by replacing each occurrence of the pattern in W with an occurrence of . Obviously, we have that $\hat{S}: D_0 \to D_1$, $\hat{F}: D_1 \to D_0$, $\hat{S}|_{\mathcal{P}} = S$, and $\hat{F}|_{Im(A)} = F$, where $\mathcal{P}$ is the set of all parallelogram polyominoes and $Im(A)$ is the image of the map $T \mapsto A(T)$. By definition of $\hat{S}$ and $\hat{F}$, we easily deduce that:

Lemma 22. $\hat{S} = \hat{F}^{-1}$.

The final step toward proving that Ψ and Λ are inverse of one another is the following lemma:

Lemma 23. *For any binary tree T, we have $A(T) = S(\Lambda(T))$.*

Proof. Let T be a binary tree. Proposition 21 states that $\Lambda(T) = F(A(T))$ is a parallelogram polyomino, hence $S(\Lambda(T))$ is well defined. Then, Lemma 22 implies that

$$S(\Lambda(T)) = S(F(A(T))) = \hat{S}(\hat{F}(A(T))) = A(T). \qquad\square$$

Proposition 24. *Ψ and Λ are inverse of one another.*

Proof. By Lemma 23, we get that $\Psi \circ \Lambda$ is the identity over the set of binary trees. This implies in particular that Ψ is surjective, and that Λ is injective. Recall from Proposition 15 that Ψ is also injective, so that we obtain that Ψ is a bijection whose inverse is Λ. $\qquad\square$

6 Perspectives

In this work, we used the notion of non-ambiguous tree for the study of various combinatorial problems. Several questions related to these objects remain open.

The first perspective concerns Bessel functions. We recall that parallelogram polyominoes are enumerated with respect to their area, width and height through the quotient of analogues of the Bessel functions J_0 and J_1 [BMV92]. This *double* appearance of Bessel functions in close contexts is striking, but unexplained for the moment. In the same vein, recall that we were able to build combinatorial objects which interpret combinatorially the integers b_k that appear in the development of J_0 – namely complete non-ambiguous trees. Is it possible to generalize this construction to coefficients of general Bessel functions J_k?

For what concerns generalizations, an intriguing question would be to define and study nice analogues of non-ambiguous trees in higher dimensions. This is indeed work in progress.

We conclude with an open problem of a more algebraic nature. The Loday-Ronco Hopf algebra $YSym$ of planar binary trees [LR98] has basis elements indexed by planar binary trees. In this setting, two permutations are in the same class when they have the same underlying tree, when coded as binary search trees. It would be interesting to study an analogous problem in which we refine these classes with respect to the underlying *non-ambiguous* tree.

Acknowledgements

References

[ABN11] J.-C. Aval, A. Boussicault, and P. Nadeau. Tree-like tableaux. In *DMTCS Proceedings 23rd International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2011)*, pages 63–74, Islande, 2011. DMTCS.

[ABN13] J.-C. Aval, A. Boussicault, and P. Nadeau. Tree-like tableaux. Full version, in preparation. 2013.

[AS64] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55 of *National Bureau of Standards Applied Mathematics Series*. U.S. Government Printing Office, Washington, D.C., 1964.

[BMV92] Mireille Bousquet-Mélou and Xavier Gérard Viennot. Empilements de segments et q-énumération de polyominos convexes dirigés. *J. Comb. Theory, Ser. A*, 60(2):196–224, 1992.

[Bur07] A. Burstein. On some properties of permutation tableaux. *Ann. Comb.*, 11(3-4):355–368, 2007.

[Car63] L. Carlitz. A sequence of integers related to the Bessel functions. *Proc. Amer. Math. Soc.*, 14:1–9, 1963.

[DDD95] M. Delest, J. P. Dubernard, and I. Dutour. Parallelogram polyominoes and corners. *J. Symbolic Comput.*, 20(5-6):503–515, 1995. Symbolic computation in combinatorics Δ_1 (Ithaca, NY, 1993).

[DV84] M.-P. Delest and G. Viennot. Algebraic languages and polyominoes enumeration. *Theoretical Computer Science*, 34(1-2):169 – 206, 1984.

[ES00] R. Ehrenborg and E. Steingrímsson. The excedance set of a permutation. *Advances in Applied Mathematics*, 24(3):284 – 299, 2000.

[Knu98] D. E. Knuth. *The art of computer programming, volume 3: (2nd ed.) sorting and searching*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998.

[LR98] Jean-Louis Loday and Maria Ronco. Hopf algebra of the planar binary trees. *Adv. Math.*, 139(2):293–309, 1998.

[Nad11] P. Nadeau. The structure of alternative tableaux. *J. Combin. Theory Ser. A*, 118(5):1638–1660, 2011.

[PWZ96] M. Petkovšek, H. S. Wilf, and D. Zeilberger. *A = B*. Ak Peters Series. Peters, 1996.

[S⁺12] W. A. Stein et al. *Sage Mathematics Software*. The Sage Development Team, 2012. `http://www.sagemath.org`.

[SCc12] The Sage-Combinat community. Sage-Combinat: enhancing Sage as a toolbox for computer exploration in algebraic combinatorics, 2012. `http://combinat.sagemath.org`.

[Slo] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences. `http://oeis.org`.

[SW07] E. Steingrímsson and L. K. Williams. Permutation tableaux and permutation patterns. *J. Combin. Theory Ser. A*, 114(2):211–234, 2007.

[Vie07] X. Viennot. Alternative tableaux, permutations and partially asymmetric exclusion process. *Talk at Isaac Newton institute*, April 2007. `http://www.newton.ac.uk/webseminars/pg+ws/2008/csm/csmw04/0423/viennot/`.