INTERVALS OF BALANCED BINARY TREES IN THE TAMARI LATTICE

SAMUELE GIRAUDO

Abstract. We show that the set of balanced binary trees is closed by interval in the Tamari lattice. We establish that the intervals $[T, T']$ where T and T' are balanced binary trees are isomorphic as posets to a hypercube. We introduce synchronous grammars that allow to generate tree-like structures and obtain fixed-point functional equations to enumerate these. We also introduce imbalance tree patterns and show that they can be used to describe some sets of balanced binary trees that play a particular role in the Tamari lattice. Finally, we investigate other families of binary trees that are also closed by interval in the Tamari lattice.

Contents

Date: January 31, 2013.
Key words and phrases. Balanced binary tree; Tamari lattice; Poset; Grammar; Generating series; Fixed-point functional equation.

1. Introduction

Binary search trees are used as data structures to represent dynamic totally ordered sets (see [AU94, Knu98, CLRS03]). The algorithms solving classical related problems such as the insertion, the deletion or the search of a given element can be performed in linear time in terms of the depth of the encoding binary tree, and, if the binary tree is *balanced*, these operations can be made in logarithmic time in terms of the cardinality of the represented set. Recall that a binary tree is balanced if for each node x, the heights of the left and the right subtrees of x differ by at most one.

The algorithmic of balanced binary trees fundamentally relies on the so-called *rotation* operation. An insertion or a deletion of an element in a dynamic ordered set modifies the binary tree encoding it and can imbalance it. The efficiency of these algorithms comes from the fact that binary search trees can be rebalanced very quickly after the insertion or the deletion, using no more than two rotations [AVL62].

Surprisingly, this operation appears in a different context since it defines a partial order on the set of binary trees of a given size. A binary tree T_0 is smaller than a binary tree T_1 if it is possible to transform T_0 into T_1 by performing a succession of right rotations. This partial order, known as the *Tamari order* [Tam62, Sta90, Knu04], defines a lattice structure on the set of binary trees of a given size.

Since binary trees are naturally equipped by this order structure induced by rotations, and the balance of balanced binary trees is maintained doing rotations, we would like to investigate if balanced binary trees play a particular role in the Tamari lattice. Our goal is to combine the two points of view of the rotation operation. Computer trials show that the intervals $[T, T']$ where T and T' are balanced binary trees are only made of balanced binary trees. The main goal of this paper is to prove this property. As a consequence, we give a characterization on the shape of these intervals and, using grammars allowing the generation of tree-like structures, enumerate these ones.

This article is organized as follows. In Section 2, we set the essential notions about binary trees and balanced binary trees, and we give the definition of the Tamari lattice in our setting. Section 3 is devoted to establish the main result: The set of balanced binary trees is closed by interval in the Tamari lattice. In Section 4, we define *synchronous grammars*. This new sort of grammars allows to generate sets of tree-like structures and gives a way to obtain a fixed-point functional equation for the generating series enumerating these. In Section 5, we introduce a notion of binary tree pattern, namely the imbalance tree patterns, and a notion of pattern avoidance. We also define subsets of balanced binary trees whose elements hold a particular position in the Tamari lattice. These sets can also be defined as the balanced binary trees avoiding some given imbalance tree patterns. In Section 6, we look at balanced binary tree intervals and show that they are, as posets, isomorphic to hypercubes. Encoding balanced binary tree intervals by kind of tree-like structures, and by constructing the synchronous grammar generating these trees, we give a fixed-point functional equation satisfied by the generating series enumerating balanced binary tree intervals. We do the same for maximal balanced binary tree intervals. Finally, in Section 7, we investigate three other families of binary trees that are closed by interval in the Tamari lattice: The weight balanced binary trees, the binary trees with a given canopy and the k-Narayana binary trees. We also look at a generalization of balanced binary trees and prove, among other, that the set of usual balanced binary trees is the only set among this generalization that is both closed by interval in the Tamari lattice and the subposet of the Tamari lattice induced by it has nontrivial intervals.

This paper is an extended version of [Gir10] where only Sections 2, 3, 5 and 6 were developed.

Acknowledgments. The author would like to thank Florent Hivert for introducing him to the problem addressed in this paper, and Jean-Christophe Novelli and Florent Hivert for their invaluable advice and their improvement suggestions. The computations of this work have been done with the open-source mathematical software Sage [S+10].

2. Preliminaries

2.1. **Complete rooted planar binary trees.** In this article, we consider complete rooted planar binary trees and we call these simply *binary trees*. Recall that a binary tree T is either a *leaf* (also called *empty tree*) denoted by $\perp$, or a node that is attached through two edges to two binary trees, called respectively the *left subtree* and the *right subtree* of T. The (unique) binary tree which has L as left subtree and R as right subtree is denoted by $L \wedge R$. Let also $\mathcal{T}_n$ be the set of binary trees with n nodes and $\mathcal{T}$ be the set of all binary trees. We use in the sequel the standard terminology (*i.e.*, *parent*, *child*, *ancestor*, *path*, *etc.*) about binary trees [AU94].

In our graphical representations, nodes are represented by circles, leaves by squares, and edges by segments or . Besides, we shall represent arbitrary subtrees by big squares like , and arbitrary paths by zigzag lines.

Recall that the *infix reading order* of the nodes of a binary tree T consists in recursively visiting its left subtree, then its root, and finally its right subtree. We say that a node x of T is the *leftmost node* if x is the first visited node in the infix order. We also say that a node y is *to the right w.r.t.* a node x if x appears strictly before y in the infix order and we denote that by $x \rightsquigarrow_T y$. We extend this notation to subtrees, saying that a subtree S of T is *to the right w.r.t.* a node x if for all nodes y of S we have $x \rightsquigarrow_T y$. For example, consider the binary tree T

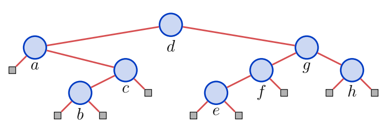
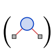
$T =$ 

Figure 1. An example of binary tree.

depicted in Figure 1. The sequence (a, b, c, d, e, f, g, h) is the sequence of all nodes of T visited in the infix order. Hence, a is the leftmost node of T and we have, among other, $a \rightsquigarrow_T b$ and $c \rightsquigarrow_T f$. Consider the subtree S of root g. It contains the nodes e, f, g and h. Hence, we have $a \rightsquigarrow_T S$, $b \rightsquigarrow_T S$, $c \rightsquigarrow_T S$ and $d \rightsquigarrow_T S$. However, we neither have the relation $e \rightsquigarrow_T S$ since S contains e, nor the relation $f \rightsquigarrow_T S$ since S contains e and $f \rightsquigarrow_T e$ does not hold.

2.2. **Balanced binary trees.** If T is a binary tree, we shall denote by $\mathrm{h}(T)$ its *height*, that is the length of the longest path connecting its root to one of its leaves. More formally,

$$\mathrm{h}(T) := \begin{cases} 1 + \max\{\mathrm{h}(L), \mathrm{h}(R)\} & \text{if } T = L \wedge R, \\ 0 & \text{otherwise } (T = \perp). \end{cases} \tag{2.1}$$

For example, we have $\mathrm{h}(\perp) = 0$, $\mathrm{h}\Big($  $\Big) = 1$, and $\mathrm{h}\Big($  $\Big) = 2$.

Let us define the *imbalance mapping* i_T which associates an element of $\mathbb{Z}$ with a node x of T. It is defined by

$$\mathrm{i}_T(x) := \mathrm{h}(R) - \mathrm{h}(L), \tag{2.2}$$

where L (resp. R) is the left (resp. right) subtree of x. For example, the imbalance values of the nodes of the binary tree T shown in Figure 1 satisfy $\mathrm{i}_T(a) = 2$, $\mathrm{i}_T(b) = 0$, $\mathrm{i}_T(c) = -1$, $\mathrm{i}_T(d) = 0$, $\mathrm{i}_T(e) = 0$, $\mathrm{i}_T(f) = -1$, $\mathrm{i}_T(g) = -1$ and $\mathrm{i}_T(h) = 0$.

A node x is *balanced* if

$$\mathrm{i}_T(x) \in \{-1, 0, 1\}. \tag{2.3}$$

Balanced binary trees form a subset of $\mathcal{T}$ composed of binary trees which have the property of being balanced:

Definition 2.1. *A binary tree T is* balanced *if all nodes of T are balanced.*

Let us denote by $\mathcal{B}_n$ the set of balanced binary trees with n nodes (see Figure 2 for the first sets) and $\mathcal{B}$ the set of all balanced binary trees. The number of balanced binary trees enumerated according to their number of nodes is Sequence **A006265** of [Slo] and begins as

$$1, 1, 2, 1, 4, 6, 4, 17, 32, 44, 60, 70, 184, 476, 872, 1553, 2720, 4288, 6312, 9004. \tag{2.4}$$

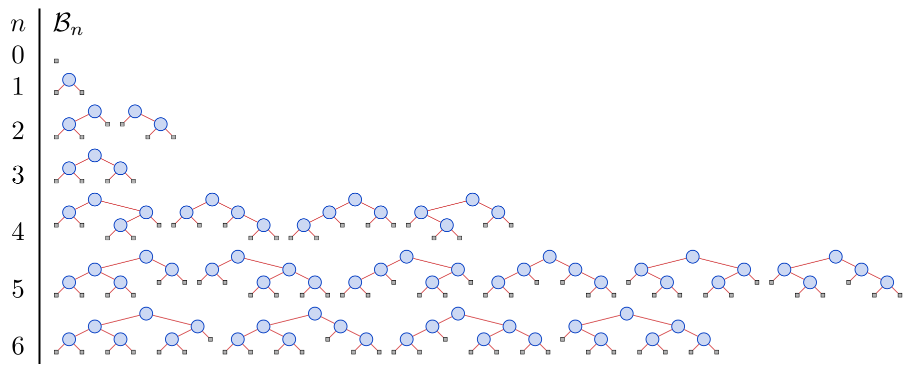


FIGURE 2. The first balanced binary trees.

2.3. **The Tamari lattice.** The Tamari lattice can be defined in several ways depending on which kind of Catalan object (*i.e.*, objects in bijection with binary trees) the order relation is defined. The most common definitions are made on integer vectors with some conditions [Sta90], on forests and binary trees [Knu04], and on Dyck paths [BB09]. We give here the most convenient definition for our use. First, let us recall the right rotation operation:

Definition 2.2. *Let T_0 be a binary tree and y be a node of T_0 having a nonempty left subtree. Let $S_0 := (A \wedge B) \wedge C$ be the subtree of root y of T_0 and T_1 be the binary tree obtained by replacing S_0 by $A \wedge (B \wedge C)$ (see Figure 3). Then the* right rotation of root y *sends T_0 to T_1.*

We write $T_0 \lessdot T_1$ if T_1 can be obtained by a right rotation from T_0. We call the relation $\lessdot$ the *partial Tamari relation*. Note that the application of a right rotation to a binary tree does not change the infix order of its nodes. In the sequel, we mainly talk about right rotations, so we call these simply *rotations*. We are now in a position to give our definition of the Tamari order.

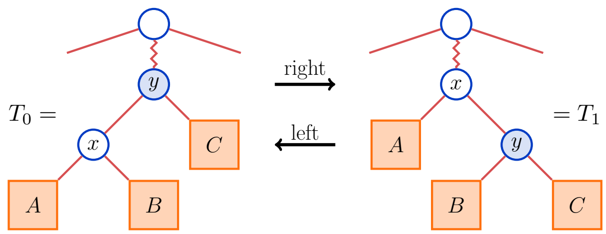


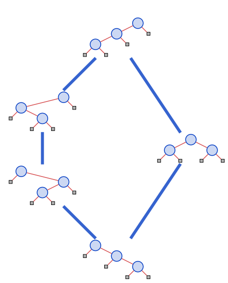
FIGURE 3. The right rotation of root y.

Definition 2.3. *The* Tamari relation $\leq_{\mathrm{T}}$ *is the reflexive and transitive closure of the partial Tamari relation* $\lessdot$. *In other words, we have* $T_0 \leq_{\mathrm{T}} T_k$ *if there exists a sequence* $T_1, \dots, T_{k-1}$ *of binary trees such that*

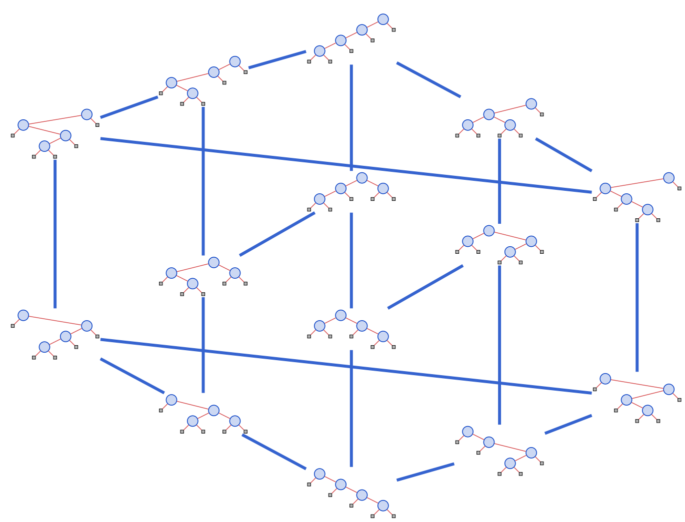
$$T_0 \lessdot T_1 \lessdot \cdots \lessdot T_{k-1} \lessdot T_k. \tag{2.5}$$

The Tamari relation is an order relation. Indeed, $\leq_{\mathrm{T}}$ is reflexive and transitive by definition. To prove that $\leq_{\mathrm{T}}$ is antisymmetric, consider the statistic $\phi : \mathcal{T} \to \mathbb{N}$ where $\phi(T)$ is the sum for all nodes x of T of the number of the nodes constituting the right subtree of x. It is plain that if $T_0 \lessdot T_1$ then $\phi(T_0) < \phi(T_1)$, showing that $\leq_{\mathrm{T}}$ is antisymmetric.

For $n \geq 0$, the set $\mathcal{T}_n$ with the order relation $\leq_{\mathrm{T}}$ defines a lattice, namely the Tamari lattice (see [HT72]). We denote by $\mathbb{T}_n := (\mathcal{T}_n, \leq_{\mathrm{T}})$ the Tamari lattice of order n (see Figure 4 for some examples).



(a) $\mathbb{T}_3$



(b) $\mathbb{T}_4$

FIGURE 4. The Tamari lattices $\mathbb{T}_3$ and $\mathbb{T}_4$. The smallest elements are at the top.

3. Closure by interval of the set of balanced binary trees

3.1. **Rotations and balance.** Let us first consider the modifications of the imbalance values of the nodes of a balanced binary tree $T_0 := (A \wedge B) \wedge C$ when a rotation at its root is applied.

Let T_1 be the binary tree obtained by this rotation, y be the root of T_0 and x be the left child of y in T_0 (see again Figure 3, considering now that y is the root of T_0 and x is the root of T_1). Note first that the imbalance values of the nodes of the subtrees A, B and C are not modified by this rotation. Indeed, only the imbalance values of x and y are changed. Since T_0 is balanced, we have $\mathrm{i}_{T_0}(x) \in \{-1, 0, 1\}$ and $\mathrm{i}_{T_0}(y) \in \{-1, 0, 1\}$. Thus, the pair $(\mathrm{i}_{T_0}(x), \mathrm{i}_{T_0}(y))$ can take nine different values. Here follows the list of the imbalance values of x and y in T_0 and T_1 expressed as $(\mathrm{i}_{T_0}(x), \mathrm{i}_{T_0}(y)) \longrightarrow (\mathrm{i}_{T_1}(x), \mathrm{i}_{T_1}(y))$:

(R1) $(-1, -1) \longrightarrow (\mathbf{1}, \mathbf{1})$,
(R4) $(1, -1) \longrightarrow (2, \mathbf{0})$,
(R7) $(-1, 1) \longrightarrow (3, 3)$,
(R2) $(0, -1) \longrightarrow (\mathbf{1}, \mathbf{0})$,
(R5) $(1, 0) \longrightarrow (3, \mathbf{1})$,
(R8) $(0, 1) \longrightarrow (3, 2)$,
(R3) $(0, 0) \longrightarrow (2, \mathbf{1})$,
(R6) $(-1, 0) \longrightarrow (2, 2)$,
(R9) $(1, 1) \longrightarrow (4, 2)$.

Let us gather these nine sorts of rotations into three different groups, taking into account if the nodes x and y are balanced in T_1.

- Cases (R1) and (R2), where x and y stay balanced are called *conservative balancing rotations*;
- Cases (R3), (R4) and (R5), where y stays balanced but x not are called *simply unbalancing rotations*;
- Cases (R6), (R7), (R8) and (R9) where x and y are both unbalanced are called *fully unbalancing rotations*.

This leads to the following properties.

Proposition 3.1. *Let T_0 and T_1 be two balanced binary trees such that $T_0 \lessdot T_1$. Then, T_0 and T_1 have the same height.*

Proof. Since T_0 and T_1 are both balanced, the rotation modifies a subtree S_0 of T_0 such that the imbalance values of the root y of S_0, and of the left child x of y, satisfy (R1) or (R2). Let S_1 be the binary tree obtained by the rotation of root y from S_0. Computing the height of S_0 and S_1, we have $\mathrm{h}(S_0) = \mathrm{h}(S_1)$. Thus, since a rotation modifies a binary tree locally, we have $\mathrm{h}(T_0) = \mathrm{h}(T_1)$. □

Lemma 3.2. *Let T_0 be a balanced binary tree and T_1 be an unbalanced binary tree such that $T_0 \lessdot T_1$. Then, there exists a node z in T_1 such that $\mathrm{i}_{T_1}(z) \geq 2$ and the left subtree and the right subtree of z are both balanced.*

Proof. Let y be the node of T_0 which is the root of the rotation that transforms T_0 into T_1 and x its left child in T_0. If this rotation is a simply unbalancing rotation, it satisfies (R3), (R4) or (R5), and the node $z := x$ satisfies the lemma. If this rotation is a fully unbalancing rotation, it satisfies (R6), (R7), (R8) or (R9), and the node $z := y$ of T_1 agrees with the conclusion of the lemma. □

Lemma 3.3. *Let T_0 be a binary tree and y be a node of T_0 such that all subtrees to the right w.r.t. y are balanced. Then, if the binary tree T_1 is obtained from T_0 by a rotation of root y, all subtrees of T_1 to the right w.r.t. y are balanced.*

Proof. Since the rotation operation does not modify the infix order of the nodes and by definition of the relation $\rightsquigarrow$, if a subtree S is to the right w.r.t. y in T_1, then S is also to the right w.r.t. y in T_0. By hypothesis, S is balanced in T_0, and therefore, it is also balanced in T_1. □

3.2. **Construction of an imbalance invariant.** Let T be a binary tree, x be a node of T and y be the leftmost node of the subtree of root x in T. We say that x is a *witness of imbalance* if the following three conditions hold (see Figure 5):

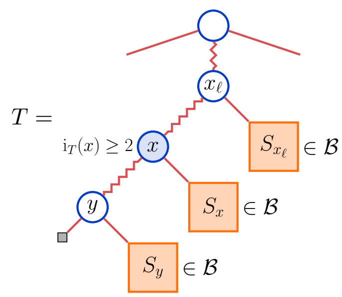


FIGURE 5. The node x is a witness of imbalance of T. Note that the left subtree of y is empty and thus S_y has 0 or 1 node.

(W1) The imbalance value of x is greater than or equal to 2;
(W2) The left subtree of x is balanced;
(W3) The subtrees of T which are to the right w.r.t. y are balanced.

Remark 3.4. *If a binary tree T has a witness of imbalance, (W1) guarantees that T is unbalanced.*

The aim of this section is to define an additional property that x and y must satisfy to ensure that any binary tree T' such that $T \leq_{\mathrm{T}} T'$ has still a witness of imbalance. In this way, by showing that T' also satisfies this additional property, we will prove that it is impossible to rebalance T through rotations.

Let us already give this property. In what follows, the concepts necessary to understand it will be defined. If y satisfies condition

(CC) the height word of the node y is admissible,

then, we say that T satisfies the *conservation condition*. Besides, we say that T has an *imbalance invariant* if T has a witness of imbalance satisfying the conservation condition.

3.2.1. *Height words.* Let T be a binary tree, x_1 be a node of T, $(x_1, x_2, \ldots, x_\ell)$ be the sequence of all ancestors of x_1 whose are to the right w.r.t. x_1 and ordered from bottom to top, and $(S_{x_i})_{1\leq i\leq \ell}$ be the sequence of the right subtrees of the x_i (see Figure 6). The word $u_1 \ldots u_\ell$ of $\mathbb{N}^*$ defined by $u_i := \mathrm{h}(S_{x_i})$ is called the *height word* of x_1 and denoted by $\mathrm{hw}_T(x_1)$. It is convenient to set $\mathrm{hw}_T(x) := \epsilon$ whenever x is not a node of T. See Figure 7 for some examples of height words associated with some nodes of a binary tree.

3.2.2. *Admissible words.* Let $u := u_1 \ldots u_n$ be a word. Let us denote by $\ell(u)$ the *length* n of u.

Let $\Theta : \mathbb{N}^2 \to \mathbb{N}$ be the rewriting rule defined by

$$\Theta(a.b) := \begin{cases} \max\{a, b\} + 1 & \text{if } b - a \in \{-1, 0, 1\}, \\ \max\{a, b\} & \text{otherwise.} \end{cases} \tag{3.1}$$

Note that if $A \wedge B$ is a balanced binary tree, then $\Theta(\mathrm{h}(A).\mathrm{h}(B)) = \mathrm{h}(A \wedge B)$. We shall use this simple observation to establish the main result of this section.

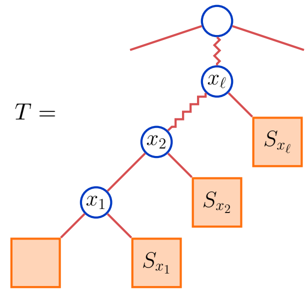


FIGURE 6. The sequence $(S_{x_i})_{1\le i\le \ell}$ associated with the node x_1.

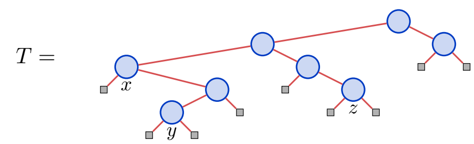


FIGURE 7. Examples of height words: $\mathrm{hw}_T(x) = 221$, $\mathrm{hw}_T(y) = 0021$, and $\mathrm{hw}_T(z) = 01$.

This rewriting rule is extended to words of $\mathbb{N}^*$ by $\Theta(u) := \Theta(u_1.u_2).u_3 \dots u_{\ell(u)}$. If $0 \le i \le \ell(u) - 1$, denote by $\Theta^i(u)$ the iterated application of Θ defined by

$$\Theta^i(u) := \begin{cases} u & \text{if } i = 0, \\ \Theta\left(\Theta^{i-1}(u)\right) & \text{otherwise.} \end{cases} \tag{3.2}$$

Definition 3.5. *A word $u \in \mathbb{N}^*$ is* admissible *if either $\ell(u) \le 1$ or all words v of the set*

$$\left\{\Theta^i(u) : 0 \le i \le \ell(u) - 2\right\} \tag{3.3}$$

satisfy $v_1 - 1 \le v_2$.

The set of admissible words is denoted by $\mathcal{A}$. To check if a word u is admissible, iteratively compute the elements of the set (3.3) following (3.2), and check for each of these the inequality of the previous definition. For example, by denoting by $\xrightarrow{\Theta}$ the rewriting rule Θ, we can check that $u := 00122$ is admissible. Indeed, we have

$$00122 \xrightarrow{\Theta} 1122 \xrightarrow{\Theta} 222 \xrightarrow{\Theta} 32, \tag{3.4}$$

and at each step, the condition $u_1 - 1 \le u_2$ holds. The word 1234488 is also admissible:

$$01233778 \xrightarrow{\Theta} 2233778 \xrightarrow{\Theta} 333778 \xrightarrow{\Theta} 43778 \xrightarrow{\Theta} 5778 \xrightarrow{\Theta} 778 \xrightarrow{\Theta} 88. \tag{3.5}$$

On the other hand, 3444 is not admissible since we have

$$3444 \xrightarrow{\Theta} 544 \xrightarrow{\Theta} 64, \tag{3.6}$$

and $6 - 1 \not\le 4$.

If u is an nonempty word, let us denote by $\Omega(u)$ the *height* of u, that is the one-letter word $\Theta^{\ell(u)-1}(u)$. For example, we have $\Omega(00122) = 4$, $\Omega(01233778) = 9$ and $\Omega(3444) = 6$. Note that one can deduce from Definition 3.5 that a word $u \in \mathbb{N}^*$ of length greater than 1 is admissible if and only if for each decomposition $u = v.a.w$ where $v \in \mathbb{N}^+$, $a \in \mathbb{N}$ and $w \in \mathbb{N}^*$, one has $\Omega(v) - 1 \le a$.

3.2.3. *Some properties of admissible words.* Let us establish three lemmas on admissible words that will be helpful later to prove our main result.

Lemma 3.6. *If u is an admissible word, then, for all $1 \leq i \leq \ell(u) - 1$, one has $u_i - 1 \leq u_{i+1}$.*

Proof. Assume that u is of the form $u = v.u_i.u_{i+1}.w$ with $v, w \in \mathbb{N}^*$ and $u_i - 1 > u_{i+1}$. Since Θ changes a word $a.b \in \mathbb{N}^2$ into a letter $c \in \mathbb{N}$ no smaller than both a and b, we have $\Omega(v.u_i) \geq u_i$. That implies that $\Omega(v.u_i) - 1 > u_{i+1}$, showing that $u \notin \mathcal{A}$ and contradicting the hypothesis. □

Lemma 3.7. *All prefixes and suffixes of an admissible word are admissible.*

Proof. It is immediate, by definition, that all prefixes of an admissible word also are admissible.

Let $u \in \mathcal{A}$ such that $\ell(u) \geq 2$, and w be a nonempty suffix of u. Assume that $w \notin \mathcal{A}$. Hence, w is of the form $w = x.a.y$ where $x \in \mathbb{N}^+$, $a \in \mathbb{N}$, $y \in \mathbb{N}^*$ and $\Omega(x) - 1 > a$. The word u is of the form $u = v.x.a.y$ where $v \in \mathbb{N}^*$. Since Θ changes a word $a.b \in \mathbb{N}^2$ into a letter $c \in \mathbb{N}$ no smaller than both a and b, we have $\Omega(v.x) \geq \Omega(x)$. Therefore, we have $\Omega(v.x) - 1 > a$, showing that $u \notin \mathcal{A}$ and contradicting the hypothesis. □

Lemma 3.8. *If $u.v$ is an admissible word such that $\ell(v) \geq 2$, the word $u.\Theta(v)$ is still admissible.*

Proof. If u is empty, the lemma follows immediately. Assume that u is nonempty. The word $u.v$ is of the form $u.v = u.a.b.w$ where $a, b \in \mathbb{N}$ and $w \in \mathbb{N}^*$. Set $c := \Theta(a.b) = \Omega(a.b)$. The word $u.c.w = u.\Theta(v)$ is admissible if the two inequalities $\Omega(u) - 1 \leq c$ and $\Omega(u.c) \leq \Omega(u.a.b)$ hold. Since $u.a.b.w \in \mathcal{A}$, we have $\Omega(u) - 1 \leq a$, and since $c = \Theta(a.b)$, then $c \geq a$ and thus, $\Omega(u) - 1 \leq c$, showing the first inequality. Set $d := \Omega(u)$. The second inequality amounts to prove that $\Omega(d.c) \leq \Omega(d.a.b)$, which is equivalent to prove $\Omega(d.\Omega(a.b)) \leq \Omega(d.a.b)$. This relation holds in general for any letters $a, b, d \in \mathbb{N}$, showing that $u.\Theta(v) \in \mathcal{A}$. □

3.2.4. *Admissible height words.* Let us prove two lemmas relating admissible words and height words.

Lemma 3.9. *Let T be a balanced binary tree, x be a node of T, and u be the height word of x. Then u is admissible and $\Omega(u) \leq \mathrm{h}(T)$.*

Proof. We proceed by structural induction on the set of balanced binary trees. The lemma is true for the single element T of the set $\mathcal{B}_1$ since by denoting x its node, we have $u = \mathrm{hw}_T(x) = 0$ which is admissible and satisfies $0 = \Omega(u) \leq \mathrm{h}(T) = 1$.

Assume that $T = L \wedge R$. If x is a node of R, we have $u = \mathrm{hw}_T(x) = \mathrm{hw}_R(x)$, and by induction hypothesis, $u \in \mathcal{A}$ and $\Omega(u) \leq \mathrm{h}(R)$. Since $\mathrm{h}(R) < \mathrm{h}(T)$, the lemma is satisfied.

If x is a node of L, we have $u = \mathrm{hw}_T(x) = \mathrm{hw}_L(x).\,\mathrm{h}(R)$. Since T is balanced, $\mathrm{h}(R) - \mathrm{h}(L) \in \{-1, 0, 1\}$, and by induction hypothesis, $\Omega\left(\mathrm{hw}_L(x)\right) \leq \mathrm{h}(L)$. Hence, $\Omega\left(\mathrm{hw}_L(x)\right) - 1 \leq \mathrm{h}(R)$. Moreover, again by induction hypothesis, $\mathrm{hw}_L(x) \in \mathcal{A}$, and hence, $u \in \mathcal{A}$. Finally, since $\Omega(u) \leq \mathrm{h}(R) + 1 \leq \mathrm{h}(T)$, the lemma is satisfied. □

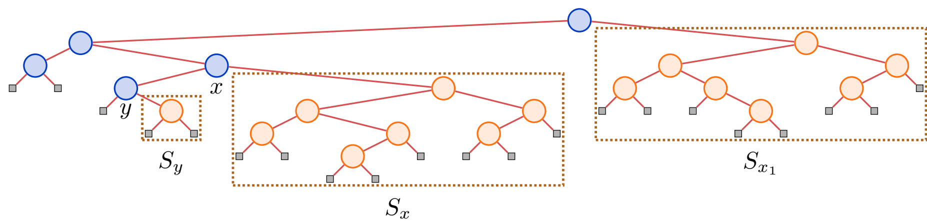
Lemma 3.10. *Let T be a binary tree and y be a node of T such that $\mathrm{hw}_T(y)$ is admissible and all subtrees of the sequence $(S_{y_i})_{1 \leq i \leq \ell}$ are balanced. Then, for all node x of T such that $y \rightsquigarrow_T x$, the word $\mathrm{hw}_T(x)$ is admissible.*

Proof. If x is an ancestor of y, since $y \rightsquigarrow_T x$, y belongs to the left subtree of x. Hence, $\mathrm{hw}_T(x)$ is a suffix of $\mathrm{hw}_T(y)$, and by Lemma 3.7, $\mathrm{hw}_T(x) \in \mathcal{A}$.

Otherwise, let S be the subtree of T such that x is a node of S and the parent of the root of S in T is an ancestor of y. The height word of y is of the form $\mathrm{hw}_T(y) = u.\,\mathrm{h}(S).v$ where $u, v \in \mathbb{N}^*$. Since $y \rightsquigarrow_T S$, by hypothesis S is balanced and thus by Lemma 3.9, $\mathrm{hw}_S(x) \in \mathcal{A}$. Thanks to Lemma 3.7, $\mathrm{h}(S).v \in \mathcal{A}$, and since, by Lemma 3.9, $\Omega(\mathrm{hw}_S(x)) \leq \mathrm{h}(S)$, the word $\mathrm{hw}_T(x) = \mathrm{hw}_S(x).v$ is admissible too. □

3.3. **The main result.** We give and prove in this section the main result of this paper. For that, we show through the next two Propositions, that the imbalance invariant defined in Section 3.2 is appropriate to prove that all successors of a binary tree obtained from a balanced binary tree by an unbalancing rotation cannot be rebalanced.

Before going further, let us give one example of a binary that satisfies the conservation condition. Let us consider the following binary tree T :

(3.7) 

.

One observes that the imbalance value of the node x is 2, that the left subtree of x is balanced, and that the subtrees to the right w.r.t. y, namely S_y, S_x, and S_{x_1} are balanced. Hence, x satisfies (W1), (W2), and (W3) and is a witness of imbalance of T. Moreover, one has $\mathrm{hw}_T(y) = 144$. Since 144 is an admissible word, T satisfies the conservation condition (CC) and hence, has an imbalance invariant.

Proposition 3.11. *Let T_0 be a balanced binary tree and T_1 be an unbalanced binary tree such that $T_0 \curlywedge T_1$. Then, T_1 has an imbalance invariant.*

Proof. Let $S_0 := (A \wedge B) \wedge C$ be the subtree of T_0 modified by the rotation transforming T_0 into T_1 and $S_1 := A \wedge (B \wedge C)$ be the resulting subtree in T_1. Denote by r the root of this rotation and by q the left child of r in S_0 (see Figure 8). We shall exhibit, in the rest of this proof, a witness of imbalance x of T_1 that satisfies the conservation condition. By Lemma 3.2,

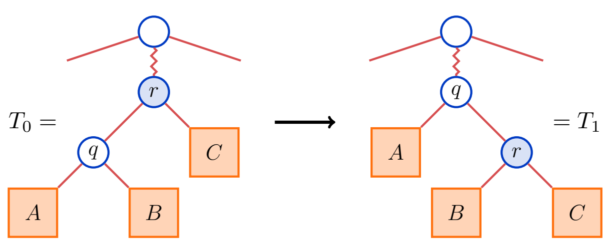


FIGURE 8. The initial case, an unbalancing rotation at root r is performed into the balanced binary tree T_0.

q or r is unbalanced in T_1 and has a positive imbalance value. Therefore, we have to consider two cases, depending on the sort of unbalancing rotation which transforms T_0 into T_1.

<u>Case 1:</u> If it is a simply unbalancing rotation, set $x := q$ and y as the leftmost node of the subtree of root q in T_1. Since $\mathrm{i}_{T_1}(x) \geq 2$, (W1) checks out. Moreover, since T_0 is balanced, by Lemma 3.3, the subtrees to the right w.r.t. r are balanced in T_1, and since A and $B \wedge C$ are balanced, (W2) and (W3) are established. Finally, since T_0 is balanced, Lemma 3.9 shows that $\mathrm{hw}_{T_0}(y)$ is admissible. We have

$$\mathrm{hw}_{T_0}(y) = \mathrm{hw}_A(y).\,\mathrm{h}(B).\,\mathrm{h}(C).v, \tag{3.8}$$

where $v \in \mathbb{N}^*$. Besides, we have

$$\mathrm{hw}_{T_1}(y) = \mathrm{hw}_A(y).\,\mathrm{h}(B \wedge C).v = \mathrm{hw}_A(y).\Theta(\mathrm{h}(B).\,\mathrm{h}(C)).v, \tag{3.9}$$

since $B \wedge C$ is balanced. Hence, we have $\mathrm{hw}_{T_1}(y) = \mathrm{hw}_A(y).\Theta(\mathrm{h}(B).\mathrm{h}(C).v)$, and since $\mathrm{hw}_A(y).\mathrm{h}(B).\mathrm{h}(C).v$ is admissible, by Lemma 3.8, $\mathrm{hw}_{T_1}(y)$ also is. That shows that (CC) is satisfied.

Case 2: Assume that the rotation is fully unbalancing. Set $x := r$ and y as the leftmost node of the subtree of root r in T_1. Since $\mathrm{i}_{T_1}(x) \geq 2$, (W1) checks out. Moreover, since T_0 is balanced, by Lemma 3.3, the subtrees to the right w.r.t r are balanced in T_1, and since B is balanced, (W2) and (W3) are established. Finally, since T_0 is balanced, Lemma 3.9 shows that $\mathrm{hw}_{T_0}(y)$ is admissible. We have

$$\mathrm{hw}_{T_0}(y) = \mathrm{hw}_B(y).\mathrm{h}(C).v, \tag{3.10}$$

where $v \in \mathbb{N}^*$. Besides,

$$\mathrm{hw}_{T_1}(y) = \mathrm{hw}_B(y).\mathrm{h}(C).v, \tag{3.11}$$

and hence $\mathrm{hw}_{T_1}(y) = \mathrm{hw}_{T_0}(y)$, so that (CC) checks out.

Thereby, we have shown that there exists a node x in T_1 that is a witness of imbalance and satisfies the conservation condition in all case. □

Proposition 3.12. *Let T_1 and T_2 be two binary trees such that $T_1 \lessdot T_2$ and T_1 has an imbalance invariant. Then, T_2 has an imbalance invariant.*

Proof. Let x be a witness of imbalance of T_1 that satisfies the conservation condition, y be the leftmost node of the subtree of root x in T_1, r be the root of the rotation that transforms T_1 into T_2, and q be the left child of r in T_1. For all relative position of r w.r.t. y in T_1, we shall exhibit a witness of imbalance x' of T_2 that satisfies the conservation condition. If necessary, we shall also exhibit the node y' of T_2 that is the leftmost node of the subtree of root x'.

There are exactly three cases to consider. Note first that since one can perform a rotation of root r, r has a left son, and since y has no left son, $r \neq y$. The first case occurs when r is to the left w.r.t. y (Case 1). Otherwise, when r is to the right w.r.t. y, the second case occurs when r is a strict ancestor of y (Case 2). In this case, y is in the left subtree of r. Otherwise, when r is to the right w.r.t. y and r is not a strict ancestor of y, the third case occurs (Case 3). In this last case, the subtree of root r is to the right w.r.t. y.

Case 1: If r is to the left w.r.t. y, the rotation of root r does not modify any of the subtrees to the right w.r.t. y. Thus, $x' := x$ is a witness of imbalance of T_2 and satisfies the conservation condition.

Case 2: If r and q are both ancestors of y in T_1, set C as the right subtree of r and B as the right subtree of q in T_1. In this case, T_2 is obtained from T_1 by replacing the subtrees B and C by $B \wedge C$ as shown in Figure 9. We have now three possibilities whether $B \wedge C$ is balanced and r is an ancestor of x in T_1.

Case 2.1: If $B \wedge C$ is unbalanced, set $x' := r$ and y' as the leftmost node of $B \wedge C$. One has

$$\mathrm{hw}_{T_1}(y) = u.\mathrm{h}(B).\mathrm{h}(C).v, \tag{3.12}$$

where $u, v \in \mathbb{N}^*$. Since x satisfies the conservation condition in T_1, $\mathrm{hw}_{T_1}(y) \in \mathcal{A}$. Thus, by Lemma 3.6, we have $\mathrm{h}(B) - 1 \leq \mathrm{h}(C)$ so that $\mathrm{i}_{T_2}(x') \geq 2$ and (W1) is satisfied. Moreover, since B is balanced, and by Lemma 3.3, all subtrees to the right w.r.t. x' are also balanced in T_2, (W2) and (W3) are established. Finally, by Lemma 3.10, $\mathrm{hw}_{T_1}(y') \in \mathcal{A}$, and since $\mathrm{hw}_{T_2}(y') = \mathrm{hw}_{T_1}(y')$, (CC) checks out.

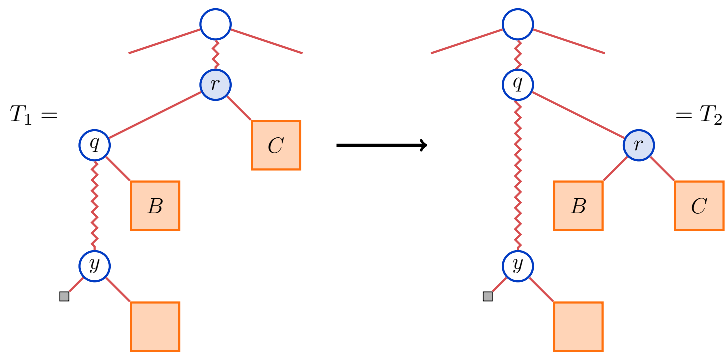


FIGURE 9. The second case, r is an ancestor of y and $y \rightsquigarrow_{T_1} r$.

Case 2.2: If $B \wedge C$ is balanced and r is an ancestor of x in T_1, set $x' := x$ and $y' := y$. One clearly has $\mathrm{i}_{T_2}(x') \geq 2$, so that (W1) is satisfied. Moreover, since the left subtree of x' in T_2 is not modified by the rotation and hence stays balanced, since $B \wedge C$ is balanced, and since by Lemma 3.3, all subtrees to the right w.r.t. r are balanced in T_2, (W2) and (W3) check out. Finally, since x satisfies the conservation condition in T_1, $\mathrm{hw}_{T_1}(y) \in \mathcal{A}$ and we have

$$\mathrm{hw}_{T_1}(y) = u.\,\mathrm{h}(B).\,\mathrm{h}(C).v, \tag{3.13}$$

where $u, v \in \mathbb{N}^*$. Besides,

$$\mathrm{hw}_{T_2}(y') = u.\,\mathrm{h}(B \wedge C).v = u.\Theta(\mathrm{h}(B).\,\mathrm{h}(C)).v, \tag{3.14}$$

since $B \wedge C$ is balanced. Thus, by Lemma 3.8, $\mathrm{hw}_{T_2}(y') \in \mathcal{A}$, so that (CC) is satisfied.

Case 2.3: If $B \wedge C$ is balanced and r is a descendant of x in T_1, we have two possibilities whether q is balanced in T_2. If it is, set $x' := x$. By Proposition 3.1, the left subtree of x' stays balanced in T_2 and $\mathrm{i}_{T_2}(x') \geq 2$. Thus, (W1) and (W2) are satisfied. Moreover, by Lemma 3.3, all subtrees to the right w.r.t. x' stay balanced in T_2 so that (W3) checks out. Otherwise, if q is not balanced, set $x' := q$. Since the left subtree of x is balanced in T_1, by Lemma 3.2, $\mathrm{i}_{T_2}(x') \geq 2$, and (W1) holds. Moreover, q belongs to the left subtree of x in T_1 which is balanced, and hence, the left subtree of q is balanced in T_2, so that (W2) holds. Since $B \wedge C$ is balanced and by Lemma 3.3, (W3) also holds. Set now for both cases y' as the leftmost node of the subtree of root x' in T_2. The word $\mathrm{hw}_{T_2}(y')$ satisfies exactly same conditions as in the previous case, so that (CC) is satisfied.

Case 3: If the subtree $S_1 := (A \wedge B) \wedge C$ of root r in T_1 is to the right w.r.t. y, set $S_2 := A \wedge (B \wedge C)$ as the subtree of T_2 obtained by the rotation at root r which transforms T_1 into T_2 (see Figure 10). We have now two cases to consider whether S_2 is balanced or not.

Case 3.1: If S_2 is balanced, by Proposition 3.1, $\mathrm{h}(S_2) = \mathrm{h}(S_1)$, and by setting $x' := x$ and $y' := y$ one has $\mathrm{i}_{T_2}(x') = \mathrm{i}_{T_1}(x)$ so that (W1) is satisfied. Moreover, the left subtree of x' stays balanced, and by Lemma 3.3, the subtrees to the right w.r.t. x' in T_2 also, so that (W2) and (W3) check out. Finally, x' also satisfies (CC) in T_2 since $\mathrm{hw}_{T_2}(y') = \mathrm{hw}_{T_1}(y)$.

Case 3.2: If S_2 is not balanced, by Proposition 3.11, there exists a node x' in S_2 which is a witness of imbalance satisfying the conservation condition, locally in S_2. Therefore, x' satisfies (W1) and (W2) in T_2. It also satisfies (W3) in T_2 since, by Lemma 3.3, the subtrees of T_2 to the right w.r.t. r stay balanced. It remains to prove that x' satisfies the conservation condition in the whole binary tree T_2. Set y' as the leftmost node of the subtree of root x' in

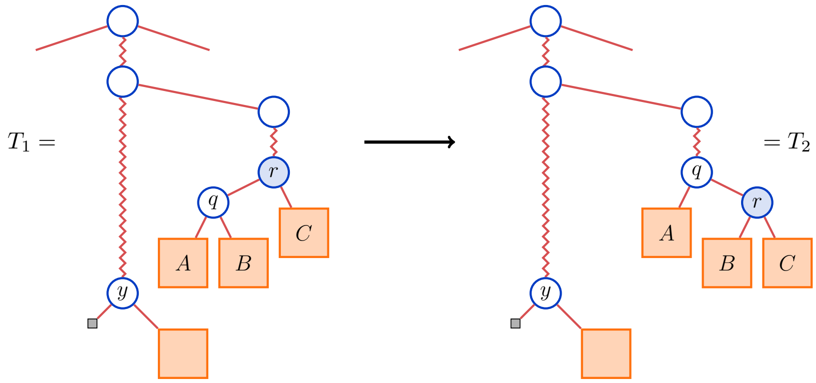


FIGURE 10. The third case, r is a node of a subtree S_1 of T_1 satisfying $y \rightsquigarrow_{T_1} S_1$.

T_2. By Proposition 3.11, $w := \mathrm{hw}_{S_2}(y') \in \mathcal{A}$, and by Lemma 3.9, w satisfies $\Omega(w) \leq \mathrm{h}(S_1)$. By hypothesis, $\mathrm{hw}_{T_1}(y) \in \mathcal{A}$ and one has

$$\mathrm{hw}_{T_1}(y) = u.\,\mathrm{h}(S_1).v, \tag{3.15}$$

where $u, v \in \mathbb{N}^*$. Besides, since

$$\mathrm{hw}_{T_2}(y') = w.v, \tag{3.16}$$

one has $\mathrm{hw}_{T_2}(y') \in \mathcal{A}$, establishing (CC).

Thereby, we have shown that there exists a node x' in T_2 that is a witness of imbalance and satisfies the conservation condition in all case. $\square$

Theorem 3.13. *Let T and T' be two balanced binary trees such that $T \leq_{\mathrm{T}} T'$. Then, the interval $[T, T']$ only contains balanced binary trees. In other words, all successors of a binary tree obtained by an unbalancing rotation from a balanced binary tree are unbalanced.*

Proof. Let T_0 and T_2 be two balanced binary trees and T_1 be an unbalanced binary tree. Assume that

$$T_0 \lessdot \cdots \lessdot T_1 \lessdot \cdots \lessdot T_2. \tag{3.17}$$

By Proposition 3.11, T_1 satisfies the conservation condition. Moreover, by Proposition 3.12, T_2 also satisfies the conservation condition. Hence, T_2 has a witness of imbalance and by Remark 3.4, T_2 is unbalanced. This is contradictory with our hypothesis.

Therefore, the notion of imbalance invariant defined in Section 3.2 is appropriate and hence the set of balanced binary trees is closed by interval in the Tamari lattice. $\square$

4. Synchronous grammars

In this section, we introduce synchronous grammars. These grammars allow to generate planar rooted tree-like structures by allowing these to grow from the root to the leaves step by step. Such trees grow from a single node, the root, and by simultaneously substituting its nodes with no children by new tree-like structures following some fixed substitution rules.

As we shall see, synchronous grammars are convenient tools to enumerate some specified families of planar rooted tree-like structures. Indeed, one can extract a fixed-point functional equation for the generating series enumerating the specified objects from a synchronous grammar subject to two precise conditions that we shall expose. We also present an algorithm to compute the coefficients of this generating series.

4.1. **Definitions.**

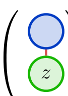
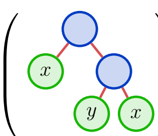
4.1.1. *Bud trees.*

Definition 4.1. *Let B be a nonempty finite alphabet. A B-*bud tree*, or simply a* bud tree *if B is fixed, is a nonempty incomplete rooted planar tree where the leaves, namely the* buds*, are labeled on B.*

Set for the sequel $B := \{\mathtt{b}_1, \ldots, \mathtt{b}_k\}$ as a nonempty finite alphabet. Denote by $\mathcal{D}_n$ the set of B-bud trees with n buds and by $\mathcal{D}$ the set of all B-bud trees. The set of all buds of a bud tree D is denoted by $\mathrm{Buds}(D)$ and the *frontier* of D is the sequence $(b_1, \ldots, b_n)$ of its buds, read from left to right. If b is a bud, we shall denote by $\mathrm{ev}(b)$ the *evaluation* of b, that is the element of B labeling b. Moreover, the *evaluation* $\mathrm{ev}(D)$ of D is the monomial of $\mathbb{Z}[B]$ defined by

$$\mathrm{ev}(D) := \prod_{b \in \mathrm{Buds}(D)} \mathrm{ev}(b). \tag{4.1}$$

For example,

(4.2) ev () $= z$ and ev () $= x^2 y$.

4.1.2. *Synchronous grammars.*

Definition 4.2. *A* synchronous grammar *S is a triple (B, a, R) where:*

- *B is a nonempty alphabet, the* set of bud labels*;*
- *a is a bud labeled on B, the* axiom *of S;*
- *$R \subseteq B \times \mathcal{D}$ is a finite set such that for all $\mathtt{b} \in B$, there is at least one bud tree D such that $(\mathtt{b}, D) \in R$. This is the set of* substitution rules *of S.*

Let $S := (B, a, R)$ be a synchronous grammar. For the sake of readability, we will make use of the following notation for substitution rules: If $(\mathtt{b}, D)$ is a substitution rule of S, we shall denote it by $\mathtt{b} \longmapsto_S D$ or by $\mathtt{b} \longmapsto D$ if S is fixed. Moreover, we will abbreviate the substitutions rules $\mathtt{b} \longmapsto_S D_1$, ..., $\mathtt{b} \longmapsto_S D_n$ by

$$\mathtt{b} \longmapsto_S D_1 + \cdots + D_n. \tag{4.3}$$

Definition 4.3. *Let $S := (B, a, R)$ be a synchronous grammar and D_0 be a bud tree with frontier $(b_1, \ldots, b_n)$ where $\mathrm{ev}(b_i) = \mathtt{b}_i$ for all $1 \leq i \leq n$. We say that the bud tree D_1 is* derivable *from D_0 in S, and we denote that by $D_0 \xrightarrow{S} D_1$, if there exists a sequence of substitution rules $(\mathtt{b}_1 \longmapsto T_1, \ldots, \mathtt{b}_n \longmapsto T_n)$ of R^n such that, by simultaneously substituting the bud b_i of D_0 by the root of T_i for all $1 \leq i \leq n$, one obtains D_1.*

Definition 4.4. *A bud tree D is* generated *by a synchronous grammar $S := (B, a, R)$ if there exists a sequence $(D_1, \ldots, D_{\ell-1})$ of bud trees such that*

$$a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D. \tag{4.4}$$

*Moreover, we say that D is generated by a ℓ-*steps derivation*.*

We denote by $\mathcal{L}_S^{(\ell)}$ the set of the bud trees generated by ℓ-steps derivations and by $\mathcal{L}_S$ the *language* of S, that is the set of all bud trees generated by S. We also say that S is *trim* if for all $\mathtt{b} \in B$ there exists at least one bud tree D generated by S that contains a bud labeled by $\mathtt{b}$. In the sequel, we shall only consider trim synchronous grammars without mentioning it explicitly.

We will illustrate most of the next definitions through the synchronous grammar

$$S_{\text{epl}} := \Big(\{x, y\}, x, R\Big), \tag{4.5}$$

where R contains the substitution rules

(4.6)

(4.7)

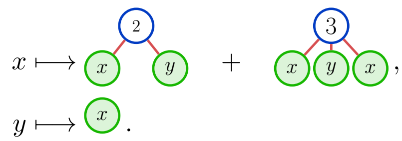


Figure 11 shows a derivation in S_{epl}.

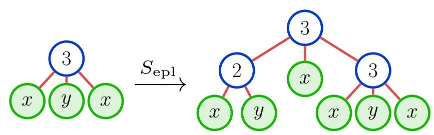


FIGURE 11. A 1-step derivation in S_{epl}.

4.1.3. *Generating graph.* The *ℓ-generating graph* $\mathcal{G}_S^{(\ell)} := (V, E)$ of a synchronous grammar S is the directed graph defined by

$$V := \bigcup_{0 \le i \le \ell} \mathcal{L}_S^{(i)}, \tag{4.8}$$

and

$$E := \left\{(D_0, D_1) \in V^2 : D_0 \xrightarrow{S} D_1\right\}. \tag{4.9}$$

The *generating graph* of S is the possibly infinite graph $\mathcal{G}_S$ defined as above where $V := \mathcal{L}_S$. This graph is connected and has at most one source, the axiom a. Figure 12 shows an example of a 2-generating graph.

4.1.4. *Strict and unambiguous synchronous grammars.*

Definition 4.5. *A synchronous grammar $S := (B, a, R)$ is* strict *if there exists a monomial order $\le_B$ on the set of monomials of $\mathbb{Z}[B]$ such that, for all bud tree D_0 generated by S and all bud tree D_1 derivable from D_0, we have $\operatorname{ev}(D_0) <_B \operatorname{ev}(D_1)$.*

If S is strict, since its set of substitution rules is finite, S generates only finitely many bud trees with a given evaluation, and since its set of buds is finite, S also generates only finitely many bud trees with a given number of buds. Moreover, if S is strict, its generating graph $\mathcal{G}_S$ is acyclic.

Lemma 4.6. *Let $S := (B, a, R)$ be a synchronous grammar. If there exists a total order $\le_B$ on B such that, for all substitution rule $\mathtt{b} \longmapsto D$ of R where $D \in \mathcal{D}_1$ we have $\mathtt{b} <_B \operatorname{ev}(D)$, then S is strict.*

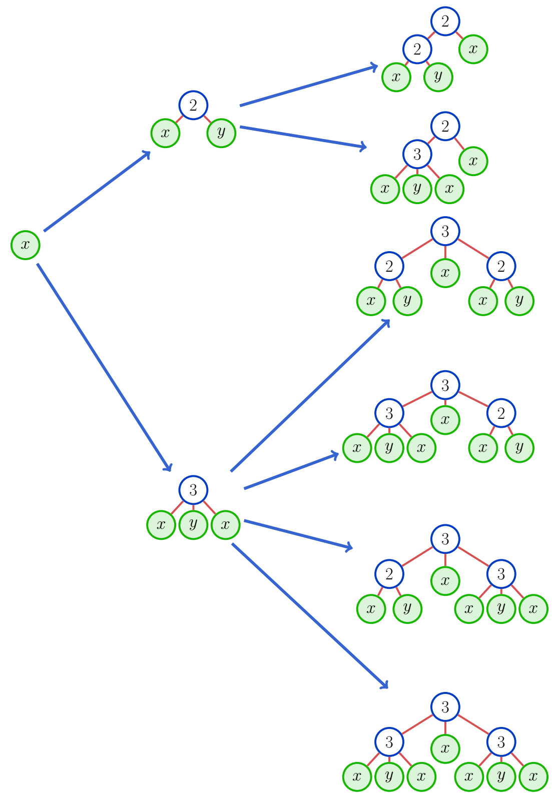


FIGURE 12. The 2-generating graph of S_{epl}.

Proof. We extend the total order $\leq_B$ defined on B into a monomial order on the set of monomials of $\mathbb{Z}[B]$ by considering the graded lexicographic order on monomials.

Consider now a bud tree D_0 generated by S and a bud tree D_1 derivable from D_0. If there exists at least one bud of D_0 that is substituted by a bud tree with more than one bud, one has $\ell(\mathrm{ev}(D_0)) < \ell(\mathrm{ev}(D_1))$ and hence $\mathrm{ev}(D_0) <_B \mathrm{ev}(D_1)$. Otherwise, D_0 and D_1 have the same number of buds. By hypothesis, all buds of the frontier $(b_1, \dots, b_n)$ of D_0 are substituted by n bud trees each containing the buds $c_1, \dots, c_n$ such that $\mathrm{ev}(b_i) <_B \mathrm{ev}(c_i)$ for all $1 \leq i \leq n$. Hence, $\mathrm{ev}(D_0) <_B \mathrm{ev}(D_1)$, implying that S is strict. □

For instance, S_{epl} is strict since the order $y \leq_B x$ meets the assumptions of Lemma 4.6. This order can be extended into the monomial order defined by

$$x^i y^j \leq_B x^k y^\ell \qquad \text{if } i + j < k + l \quad \text{or} \quad i + j = k + l \text{ and } i \leq k. \tag{4.10}$$

Definition 4.7. *A synchronous grammar S is* unambiguous *if for all bud tree D, there exists at most one integer $\ell \geq 0$ and one sequence $(D_1, \dots, D_{\ell-1})$ such that (4.4) holds.*

The generating graph $\mathcal{G}_S$ is a tree if and only if S is unambiguous.

Lemma 4.8. *Let $S := (B, a, R)$ be a strict synchronous grammar. If for all $\mathtt{b} \in B$ and for all substitution rules $\mathtt{b} \longmapsto T_0$ and $\mathtt{b} \longmapsto T_1$ of R where $T_0 \neq T_1$ there are at the same location in T_0 and T_1 two non-bud nodes that are different, then S is unambiguous.*

Proof. Let D be a bud tree generated by S and D_0 and D_1 be two different bud trees derivable from D. Among other substitutions, the bud tree D_0 (resp. D_1) is obtained by replacing one of its buds by a bud tree T_0 (resp. T_1), and by hypothesis, there are at the same location in T_0 and T_1 two non-bud nodes that are different. Hence, there are at the same location in D_0 and D_1 two non-bud nodes that are different. This shows that all bud trees obtained by performing any sequence of derivations from D_0 and from D_1 are different since they differ by a non-bud node. Moreover, since S is strict, its generating graph contains no cycle, and hence, S is unambiguous. □

For instance, Lemma 4.8 shows that S_{epl} is unambiguous since it is strict and the bud x can be substituted by two buds trees with different roots: One of these is of arity 2 while the other one is of arity 3.

4.2. Synchronous grammars and generating series.

Definition 4.9. *Let $S := (B, a, R)$ be a synchronous grammar. The* ℓ-generating series $\mathcal{S}_S^{(\ell)}$ *of S is the polynomial of $\mathbb{Z}[B]$ defined by*

$$\mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k) := \sum_{a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_\ell} \mathrm{ev}(D_\ell). \tag{4.11}$$

Moreover, if S is strict, the generating series $\mathcal{S}_S$ *of S is the element of $\mathbb{Z}[[B]]$ defined by*

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) := \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k). \tag{4.12}$$

Let $S := (B, a, R)$ be a strict synchronous grammar. The series $\mathcal{S}_S$ is well-defined since S is strict. Moreover, if S is also unambiguous, we have

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) = \sum_{D \,\in\, \mathcal{L}_S} \mathrm{ev}(D), \tag{4.13}$$

and for all monomial $u := \mathtt{b}_1^{\alpha_1} \dots \mathtt{b}_k^{\alpha_k}$, the coefficient $[u]\mathcal{S}_S$ is the number of bud trees generated by S that have u as evaluation, *i.e.*, a frontier made of α_i occurrences of buds labeled by $\mathtt{b}_i$, for all $1 \leq i \leq k$.

For example, the first ℓ-generating series of S_{epl} are

$$\mathcal{S}_{S_{\mathrm{epl}}}^{(0)}(x, y) = x, \tag{4.14}$$

$$\mathcal{S}_{S_{\mathrm{epl}}}^{(1)}(x, y) = xy + x^2 y, \tag{4.15}$$

$$\mathcal{S}_{S_{\mathrm{epl}}}^{(2)}(x, y) = x^2 y + x^3 y + x^3 y^2 + 2x^4 y^2 + x^5 y^2. \tag{4.16}$$

and its generating series is of the form

$$\mathcal{S}_{S_{\mathrm{epl}}}(x, y) = x + xy + 2x^2 y + x^3 y + x^3 y^2 + 2x^4 y^2 + x^5 y^2 + \cdots \tag{4.17}$$

For all $\mathtt{b} \in B$ let us define the polynomials $\mathrm{subs}(\mathtt{b})$ of $\mathbb{Z}[B]$ by

$$\mathrm{subs}(\mathtt{b}) := \sum_{(\mathtt{b}, D) \,\in\, R} \mathrm{ev}(D). \tag{4.18}$$

For instance, for S_{epl} one directly obtains from (4.6) and (4.7)

$$\mathrm{subs}(x) = xy + x^2 y, \tag{4.19}$$

$$\mathrm{subs}(y) = x. \tag{4.20}$$

Lemma 4.10. *Let $S := (B, a, R)$ be a synchronous grammar. For all $\ell \geq 0$, $\mathcal{S}_S^{(\ell)}$ satisfies*

$$\mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k) = \begin{cases} \operatorname{ev}(a) & \text{if } \ell = 0, \\ \mathcal{S}_S^{(\ell-1)}(\operatorname{subs}(\mathtt{b}_1), \dots, \operatorname{subs}(\mathtt{b}_k)) & \text{otherwise.} \end{cases} \tag{4.21}$$

Proof. If $\ell = 0$, the only bud tree generated by 0-step derivations is the axiom a of S. Hence, the lemma is satisfied.

Let $\ell \geq 1$. Assume that there exists the following sequence of derivations in S:

$$a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D_\ell. \tag{4.22}$$

Then, by definition, D_ℓ is obtained by substituting the buds b_i of $D_{\ell-1}$ by some buds trees T_i. From the polynomial point of view, the monomial $\operatorname{ev}(D_\ell)$ is obtained by the polynomial substitutions $\operatorname{ev}(b_i) \hookleftarrow \operatorname{ev}(T_i)$ in $\mathcal{S}_S^{(\ell-1)}$. Hence, $\mathcal{S}_S^{(\ell)}$ is obtained from $\mathcal{S}_S^{(\ell-1)}$ by performing the polynomial substitution $\mathtt{b} \leftarrow \operatorname{subs}(\mathtt{b})$ for each $\mathtt{b} \in B$, showing (4.21). □

Proposition 4.11. *Let $S := (B, a, R)$ be a strict synchronous grammar. The generating series $\mathcal{S}_S$ satisfies the fixed-point functional equation*

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) = \operatorname{ev}(a) + \mathcal{S}_S(\operatorname{subs}(\mathtt{b}_1), \dots, \operatorname{subs}(\mathtt{b}_k)). \tag{4.23}$$

Proof. Using Lemma 4.10, we obtain

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) = \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k) \tag{4.24}$$

$$= \mathcal{S}_S^{(0)}(\mathtt{b}_1, \dots, \mathtt{b}_k) + \sum_{\ell \geq 1} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k) \tag{4.25}$$

$$= \operatorname{ev}(a) + \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell+1)}(\mathtt{b}_1, \dots, \mathtt{b}_k) \tag{4.26}$$

$$= \operatorname{ev}(a) + \sum_{\ell \geq 0} \mathcal{S}_S^{(\ell)}(\operatorname{subs}(\mathtt{b}_1), \dots, \operatorname{subs}(\mathtt{b}_k)) \tag{4.27}$$

$$= \operatorname{ev}(a) + \mathcal{S}_S(\operatorname{subs}(\mathtt{b}_1), \dots, \operatorname{subs}(\mathtt{b}_k)).$$

□

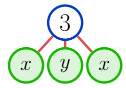
Proposition 4.11 gives a formula to extract a fixed-point functional equation for the generating series of a given strict synchronous grammar $S := (B, a, R)$ and Lemma 4.10 gives an algorithm to compute its coefficients by *iteration*, *i.e.*, by computing the polynomials $\mathcal{S}_S^{(\ell)}$ for $0 \leq \ell \leq n$ where n is a desired order, and then, by summing its terms.

In our example, the generating series of S_{epl} satisfies the fixed-point functional equation

$$\mathcal{S}_{S_{\mathrm{epl}}}(x, y) = x + \mathcal{S}_{S_{\mathrm{epl}}}(xy + x^2y, x). \tag{4.28}$$

Note that in some cases it is useful to specialize the generating series $\mathcal{S}_S$ associated with S. For example, the specialization of an element $\mathtt{b}$ of B to 0 allows to annihilate some terms of $\mathcal{S}_S$ corresponding to bud trees which have buds labeled by $\mathtt{b}$. In this way, the enumeration provided by $\mathcal{S}_S$ with this specialization takes into account only bud trees generated by S that have no bud labeled by $\mathtt{b}$.

In the same way, it is possible to add some parameters to the substitution rules of S in order to refine the generating series $\mathcal{S}_S$. For instance, to take into account the number of application of the substitution rule

(4.29) $x \longmapsto$  ,

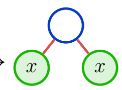
in the bud trees generated by S_{epl}, one has just to set

$$\mathrm{subs}(x) := xy + x^2 y \xi, \tag{4.30}$$

so that the parameter ξ counts the number of application of this substitution rule. In this way, one can enumerate tree-like structures according to some statistics.

4.3. **Examples.** Let us consider three examples of synchronous grammars to illustrate the concepts that we have presented. Let us start with a very simple example.

4.3.1. *Perfect binary trees.* Let the synchronous grammar $S_{\mathrm{perf}} := \left(\{x\}, x, R\right)$ where R contains the unique substitution rule

(4.31) $x \longmapsto$  .

By identifying the buds x with leaves, the language $\mathcal{L}_{S_{\mathrm{perf}}}$ is the set of *perfect binary trees*, that are binary trees of the sequence $(T_i)_{i\geq 0}$ defined by $T_0 :=\perp$ and $T_{i+1} := T_i \wedge T_i$.

This synchronous grammar is strict since the number of buds of all bud trees generated by S_{perf} increases after each derivation. Besides, since S_{perf} is strict and R only contains one substitution rule, the generating graph $\mathcal{G}_{S_{\mathrm{perf}}}$ only contains one maximal path and hence, S_{perf} is unambiguous. Therefore, the series $\mathcal{S}_{S_{\mathrm{perf}}}$ is well-defined and by Proposition 4.11, it satisfies the fixed-point functional equation

$$\mathcal{S}_{S_{\mathrm{perf}}}(x) = x + \mathcal{S}_{S_{\mathrm{perf}}}(x^2), \tag{4.32}$$

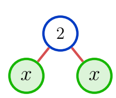
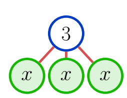
and enumerate perfect binary trees according to their number of leaves. First $\mathcal{S}^{(\ell)}_{S_{\mathrm{perf}}}$ polynomials are

$$\mathcal{S}^{(0)}_{S_{\mathrm{perf}}}(x) = x, \qquad \mathcal{S}^{(1)}_{S_{\mathrm{perf}}}(x) = x^2, \qquad \mathcal{S}^{(2)}_{S_{\mathrm{perf}}}(x) = x^4, \qquad \mathcal{S}^{(3)}_{S_{\mathrm{perf}}}(x) = x^8, \tag{4.33}$$

so that

$$\mathcal{S}_{S_{\mathrm{perf}}}(x) = \sum_{n\geq 0} x^{2^n} = x + x^2 + x^4 + x^8 + \cdots \tag{4.34}$$

4.3.2. *Balanced 2-3 trees.* Let the synchronous grammar $S_{23} := \left(\{x\}, x, R\right)$ where R contains the substitution rules

(4.35) $x \longmapsto$  $+$  .

By identifying the buds x with leaves, the language of S_{23} is the set of *balanced 2-3 trees*, that are complete rooted planar trees such that each internal node has 2 or 3 children and all paths leading to their leaves have same length (see [Odl82, FS09]).

Since each step of derivation increases the number of buds of the generated bud tree, S_{23} is strict. Moreover, S_{23} satisfies the hypothesis of Lemma 4.8 and hence is unambiguous. Indeed, the two bud trees appearing in the two substitution rules have a different root: One is of arity

2 and the other of arity 3. Thus, the series $\mathcal{S}_{S_{23}}$ is well-defined and by Proposition 4.11, it satisfies the fixed-point functional equation

$$\mathcal{S}_{S_{23}}(x) = x + \mathcal{S}_{S_{23}}(x^2 + x^3), \tag{4.36}$$

and enumerate balanced 2-3 trees according to their number of leaves. First polynomials $\mathcal{S}^{(\ell)}_{S_{23}}$ are

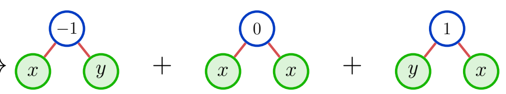
$$\mathcal{S}^{(0)}_{S_{23}}(x) = x, \tag{4.37}$$

$$\mathcal{S}^{(1)}_{S_{23}}(x) = x^2 + x^3, \tag{4.38}$$

$$\mathcal{S}^{(2)}_{S_{23}}(x) = x^4 + 2x^5 + 2x^6 + 3x^7 + 3x^8 + x^9. \tag{4.39}$$

An interpretation of the polynomial $\mathcal{S}^{(2)}_{S_{23}}(x)$ is the following: By performing 2-steps derivations, S_{23} generates one bud tree with 4 buds, two bud trees with 5 buds, two bud trees with 6 buds, three bud trees with 7 buds, three bud trees with 8 buds and one bud tree with 9 buds.

4.3.3. *Balanced binary trees.* Consider now the synchronous grammar $S_{\text{bal}} := \left(\{x, y\}, x, R\right)$ where R contains the substitution rules

(4.40) $x \longmapsto$ 

$$y \longmapsto x. \tag{4.41}$$

As we shall show, by annihilating the bud trees containing some buds y and by replacing the buds x by leaves, the language of S_{bal} is the set of balanced binary trees.

Proposition 4.12. *Let D be a bud tree generated by S_{bal} only containing buds x. Then, the non-bud nodes of D are labeled by their imbalance value.*

Proof. Each step of derivation leading to D substitutes each x by new bud trees of height two, and each y by new bud trees of height one. Thus, each step of derivation increases by one the height of the subtrees containing a x. Besides, the role of the y is to delay, during one step of derivation, the growth of the branch containing these, to enable the creation of the imbalance values -1 and 1. Since D does not have any y, every growing delay is respected, so that imbalance values are its labels. □

Proposition 4.12 shows that the bud trees generated by S_{bal} only containing buds x are balanced binary trees. Moreover, a simple structural induction on balanced binary trees shows that every balanced binary tree can be generated by S_{bal}. Indeed, the empty tree can be generated, and, if T is a balanced binary tree and z its root, by induction hypothesis, its left subtree and its right subtree can be generated by S_{bal}. To generate T, one just have to make the first derivation according to the imbalance value of z. Figure 13 shows an example of generation of a balanced binary tree.

By setting $y \leq_B x$, S_{bal} satisfies the hypothesis of Lemma 4.6 and hence, is strict. Moreover, Lemma 4.8 shows that S_{bal} is unambiguous since all bud trees arising in a right member of the substitution rules of S_{bal} have a different root since their labeling differ. Proposition 4.12 shows that this labeling is consistent. Hence $\mathcal{S}_{S_{\text{bal}}}$ is well-defined. By Proposition 4.11, the generating series enumerating the elements of $\mathcal{L}_{S_{\text{bal}}}$ satisfies the fixed-point functional equation

$$\mathcal{S}_{S_{\text{bal}}}(x, y) = x + \mathcal{S}_{S_{\text{bal}}}(x^2 + 2xy, x). \tag{4.42}$$

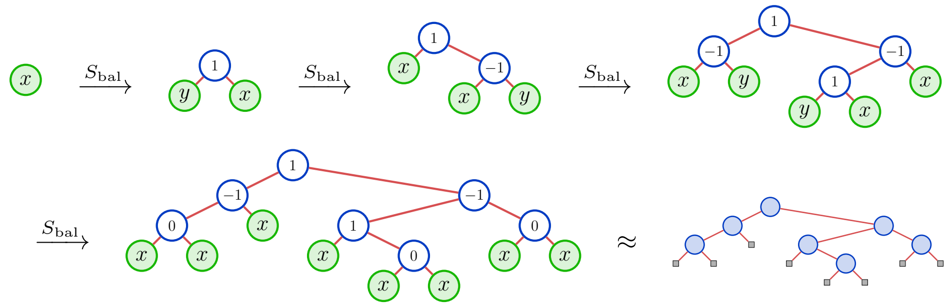


FIGURE 13. Generation of a balanced binary tree by the synchronous grammar S_{bal}.

First $\mathcal{S}^{(\ell)}_{S_{\mathrm{bal}}}$ polynomials are

$$\mathcal{S}^{(0)}_{S_{\mathrm{bal}}}(x,y) = x, \tag{4.43}$$

$$\mathcal{S}^{(1)}_{S_{\mathrm{bal}}}(x,y) = 2xy + x^2, \tag{4.44}$$

$$\mathcal{S}^{(2)}_{S_{\mathrm{bal}}}(x,y) = 4x^2y + 2x^3 + 4x^2y^2 + 4x^3y + x^4. \tag{4.45}$$

As already mentioned, to enumerate balanced binary trees, we have to discard the elements of $\mathcal{L}_{S_{\mathrm{bal}}}$ that contain a bud labelled by y. Thus, the generating series enumerating balanced binary trees according to their number of leaves is given by the specialization $\mathcal{S}_{S_{\mathrm{bal}}}(x,0)$. Note that this fixed-point functional equation is obtained in [BLL88, BLL94, Knu98] by other methods.

5. Imbalance tree patterns and balanced binary trees

Word patterns and permutations patterns are usually used to describe languages or sets of permutations by considering the set of elements avoiding these ones. We use the same idea to describe sets of binary trees by introducing a notion of binary tree patterns and pattern avoidance.

We show that we can describe in this way some interesting subsets of the set of balanced binary trees according to their particular position in the Tamari lattice. Afterwards, we use the methods developed in the previous section to construct synchronous grammar generating the maximal balanced binary trees in the Tamari lattice and get fixed-point functional equation for the generating series enumerating these.

5.1. **Imbalance tree patterns.**

Definition 5.1. *An* imbalance tree pattern *is a nonempty incomplete rooted planar binary tree with labels in* $\mathbb{Z}$.

Let T be a binary tree. We denote by T^{i} the labeled binary tree of shape T whose nodes are labeled by their imbalance value. We say that T admits an *occurrence* of the imbalance tree pattern p if a connected component of T^{i} has the same shape and same labels as p.

Now, given a set P of imbalance tree patterns, we can define the set composed of the binary trees that *avoid* P, *i.e.*, the binary trees that do not admit any occurrence of the elements of P. For example, the set

$$\left\{ i : i \notin \{-1, 0, 1\} \right\} \tag{5.1}$$

describes the set of balanced binary trees, the set

$$\left\{ i : i \neq 0 \right\} \tag{5.2}$$

describes the set of perfect binary trees and

$$\left\{ \begin{smallmatrix} & j \\ i & \end{smallmatrix} : i, j \in \mathbb{Z} \right\} \tag{5.3}$$

describes the set of *right comb binary trees*, that are binary trees such that each node has an empty left subtree.

As exposed in Section 4.3, synchronous grammars allow to generate binary trees by controlling the imbalance values of their nodes. Hence, they allow to generate binary trees that avoid some imbalance tree patterns.

5.2. **Minimal and maximal balanced binary trees in the Tamari lattice.**

5.2.1. *Minimal and maximal balanced binary trees.* Let us first describe a set of balanced binary trees and its counterpart whose elements are, roughly speaking, at the end of the balanced binary trees subposet of the Tamari lattice.

Definition 5.2. *A balanced binary tree T_0 (resp. T_1) is* maximal *(resp.* minimal*) if, for all binary tree T_1 (resp. T_0) such that $T_0 \lessdot T_1$, we have T_1 (resp. T_0) unbalanced.*

By Theorem 3.13, if T_0 (resp. T_1) is a maximal (resp. minimal) balanced binary tree, then there does not exist any balanced binary tree T_1 (resp. T_0) such that $T_0 \leq_{\mathrm{T}} T_1$. Maximal (resp. minimal) balanced binary trees are thus maximal (resp. minimal) elements in the Tamari lattice restricted to balanced binary trees.

Proposition 5.3. *A balanced binary tree T is maximal if and only if it avoids the set of imbalance tree patterns*

$$P_{\max} := \left\{ \begin{smallmatrix} & -1 \\ -1 & \end{smallmatrix}, \quad \begin{smallmatrix} & -1 \\ 0 & \end{smallmatrix} \right\}. \tag{5.4}$$

Similarly, a balanced binary tree T is minimal if and only if it avoids the set of imbalance tree patterns

$$P_{\min} := \left\{ \begin{smallmatrix} 1 & \\ & 1 \end{smallmatrix}, \quad \begin{smallmatrix} 1 & \\ & 0 \end{smallmatrix} \right\}. \tag{5.5}$$

Proof. Assume that T is maximal. Then, for all binary tree T_1 such that $T \lessdot T_1$ we have T_1 unbalanced. Thus, it is impossible to do a conservative balancing rotation into T and, looking at the different sorts of rotations studied in Section 3.1 it avoids the set $P_{\max}$.

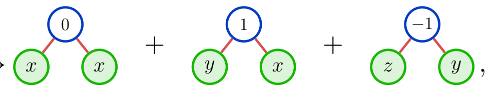
Conversely, assume that T is a balanced binary tree that avoids the two patterns of $P_{\max}$, then, for every binary tree T_1 such that $T \lessdot T_1$, T_1 is unbalanced since for all node y which has a left child x in T, the imbalance values of x and y satisfy one of the seven cases (R3)–(R9) of Section 3.1. Thus, we can only do unbalancing rotations into T, implying that T is maximal.

The second part of the proposition is done in an analogous way, considering left rotations instead of right rotations. □

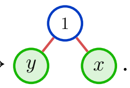
Proposition 5.4. *The generating series enumerating maximal balanced binary trees according to the number of leaves of the trees is $\mathcal{S}_{\max}(x, 0, 0)$ where*

$$\mathcal{S}_{\max}(x, y, z) = x + \mathcal{S}_{\max}(x^2 + xy + yz, x, xy). \tag{5.6}$$

Proof. To obtain this fixed-point functional equation, let us consider the synchronous grammar $S_{\max} := \left(\{x, y, z\}, x, R\right)$ where R contains the substitution rules

(5.7) $x \longmapsto$ 

(5.8) $y \longmapsto x,$

(5.9) $z \longmapsto$ 

We can apply the same idea developed in the proof of Proposition 4.12 to show that the bud trees generated by $S_{\max}$ that only contain buds x have non-bud nodes labeled by their imbalance values. Hence, by identifying in such trees the x with leaves, $S_{\max}$ only generates maximal balanced binary trees. Indeed, by Proposition 5.3, the generated trees must avoid the two patterns of $P_{\max}$. To do that, we have to control the growth of the x when they are substituted by bud trees D whose roots have imbalance values of -1. Indeed, if the root of the left subtree of D grows with an imbalance value of -1 or 0, one of the two patterns is not avoided. The idea is to force the imbalance value of the root of the left subtree of D to be 1, role played by the bud z. The role of the bud y is to delay the growth of a branch of the generated bud tree in order to create the imbalance values -1 and 1. Moreover, by structural induction on maximal balanced binary trees, one can also prove that all maximal balanced binary trees can be generated by $S_{\max}$.

By setting $y \leq_B x$, $S_{\max}$ satisfies the hypothesis of Lemma 4.6, and hence, is strict. This synchronous grammar is also unambiguous since it satisfies the hypothesis of Lemma 4.8. Indeed, the roots of all bud trees appearing in a right member of the substitution rules of R are different to one other, due to their labeling.

Finally, since $S_{\max}$ is strict and unambiguous, by Proposition 4.11, we obtain the stated fixed-point functional equation, and the generating series is obtained by the specialization $y = 0$ and $z = 0$ in order to ignore bud trees containing a bud labelled by y or by z. ☐

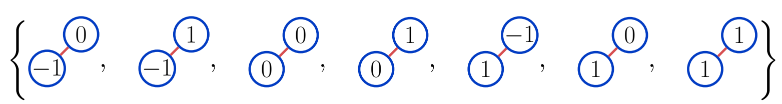
The solution of this fixed-point functional equation gives us the following first values for the number of maximal balanced binary trees in the Tamari lattice:

(5.10)
$$1, 1, 1, 1, 2, 2, 2, 4, 6, 9, 11, 13, 22, 38, 60, 89, 128, 183, 256, 353, 512, 805, 1336, 2221, 3594, \\ 5665, 8774, 13433, 20359, 30550, 45437, 67086, 98491, 144492, 213876.$$

5.2.2. *Interior balanced binary trees.* Let us now describe a set of balanced binary trees and its counterpart whose elements are, roughly speaking, in the heart of the balanced binary trees subposet of the Tamari lattice.

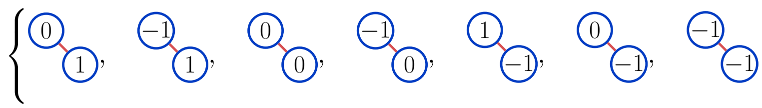
Definition 5.5. *A balanced binary tree T_0 (resp. T_1) is* right interior *(resp.* left interior*) if all binary tree T_1 (resp. T_0) such that $T_0 \lessdot T_1$ is balanced.*

Proposition 5.6. *A balanced binary tree T is right interior if and only if it avoids the set of imbalance tree patterns*

(5.11) $P_{\mathrm{rint}} :=$ 

Similarly, a balanced binary tree T is left interior if and only if it avoids the set of imbalance tree patterns

(5.12) $$P_{\mathrm{lint}} :=$$



Proof. Assume that T is right interior. Then, for all binary tree T_1 such that $T \nwarrow T_1$, T_1 is balanced. Thus, for every node y and its left child x in T, the imbalance values of x and y satisfy (R1) or (R2) of Section 3.1 since one can only do conservative balancing rotations in T. Hence, T must avoid the seven given patterns.

Conversely, assume that T is a balanced binary tree that avoids the patterns of P_{rint}. For every node y which has a left child x in T, the imbalance values of x and y satisfy (R1) or (R2). Thus, the rotation of root y in T produces a balanced binary tree and implies that T is interior.

The second part of the proposition is done in an analogous way, considering left rotations instead of right rotations. □

In the sequel, we shall only consider right interior balanced binary trees so we call these *interior* balanced binary trees. This family of binary trees is easily enumerable according to their height:

Proposition 5.7. *The number a_h of interior balanced binary trees of height h is*

(5.13) $$a_h = \begin{cases} 1 & \text{if } h \in \{0, 1, 3\}, \\ 2 & \text{if } h = 2, \\ a_{h-1}a_{h-2} & \text{otherwise.} \end{cases}$$

Proof. The values of a_h for $0 \leq h \leq 3$ can easily be computed by hand.

Let us first observe that if $T := L \wedge R$ is an interior balanced binary tree of height $h \geq 3$, then L and R also are interior balanced binary trees and the imbalance value of the root of T is -1. Indeed, if L or R is not an interior balanced binary tree, then, by Proposition 5.6, L or R would admit an occurrence of a pattern of P_{rint} and hence, would T. Moreover, if the imbalance value of T is not -1, since T is balanced and $\mathrm{h}(T) \geq 3$, its left subtree L is nonempty and T would admit an occurrence of a pattern of P_{rint}.

Let us finally show that for all integer $h \geq 4$ and all interior balanced binary trees L and R such that $\mathrm{h}(L) = h - 1$ and $\mathrm{h}(R) = h - 2$, the binary tree $T := L \wedge R$ is an interior balanced binary tree. Since $\mathrm{h}(L) \geq 3$, according to what we have just shown, the imbalance value of the root x of L is -1. The imbalance value of the root y of T also is -1 and thus, x and y do not form a pattern of P_{rint} in T. Moreover, the root of R and the node x in T do neither form a pattern of P_{rint}. Hence, T is an interior balanced binary tree. That proves (5.13). □

The first values of $(a_h)_{h \geq 0}$ are

(5.14) $$1, 1, 2, 1, 2, 2, 4, 8, 32, 256, 8192, 2097152, 17179869184.$$

By forgetting the first three values, this is Sequence **A000301** of [Slo]. Moreover, one has $a_h = 2^{f_{h-3}}$ for all $h \geq 3$, where f_i is the i-th Fibonacci number, defined by $f_i := i$ if $i \in \{0, 1\}$, and $f_i := f_{i-1} + f_{i-2}$ otherwise.

Recall that the set of *Fibonacci binary trees* [CLRS03] is formed of the elements of the sequence $(T_i)_{i \geq 0}$ where $T_0 := T_1 := \perp$ and $T_{i+2} := T_{i+1} \wedge T_i$. One can prove by structural induction on the set of Fibonacci binary trees that these also are interior balanced binary trees.

5.2.3. *Mixed balanced binary trees.* Let us finally characterize balanced binary trees which are neither maximal nor interior.

Definition 5.8. *A balanced binary tree T_0 is* right mixed *(resp.* left mixed*) if there exists a balanced binary tree T_1 and an unbalanced binary tree T'_1 such that $T_0 \lessdot T_1$ and $T_0 \lessdot T'_1$ (resp. $T_1 \lessdot T_0$ and $T'_1 \lessdot T_0$).*

Proposition 5.9. *A balanced binary tree T is right mixed (resp. left mixed) if and only if it admits at least one occurrence of an imbalance tree pattern of the set $P_{\max}$ (resp. $P_{\min}$) and at least one occurrence of an imbalance tree pattern of the set P_{rint} (resp. P_{lint}).*

Proof. Assume that T is a mixed balanced binary tree. By definition, it is possible to perform a conservative balancing rotation into T. Hence, there are two nodes x and y in T satisfying (R1) or (R2) of Section 3.1 and form an occurrence of a pattern of $P_{\max}$. Moreover, again by definition, it is possible to perform an unbalancing rotation into T. Hence, there are two nodes x' and y' in T satisfying one of the seven cases (R3)–(R9) of Section 3.1 and form an occurrence of a pattern of P_{rint}.

Conversely, if T admits some occurrences of patterns of both $P_{\max}$ and P_{rint}, considering the nine cases of rotation in a balanced binary tree studied in Section 3.1, we see that it is possible to make both a conservative and an unbalancing rotation into T, and hence T is a right mixed balanced binary tree.

The second part of the proposition is done in an analogous way, considering left rotations instead of right rotations. □

In the sequel, we shall only consider right mixed balanced binary trees, so we call these *mixed* balanced binary trees.

Note that, for $n \geq 3$, the set $\mathcal{B}_n$ is a disjoint union of the set M of maximal balanced binary trees, the set N of interior balanced binary trees and the set X of mixed balanced binary trees with n nodes. Indeed, by definition, M and X are disjoint, and in the same way, N and X also are. Consider now a balanced binary tree T which is both maximal and interior. That implies that T is the maximal element of its Tamari lattice, and hence, T is a right comb binary tree. Since T is also balanced, it cannot have more than two nodes.

6. The subposet of the Tamari lattice of balanced binary trees

6.1. **Isomorphism between balanced binary tree intervals and hypercubes.**

Lemma 6.1. *Let T_0 and T_1 be two balanced binary trees such that $T_0 \leq_{\mathrm{T}} T_1$ and y be a node of T_0. Then:*

- *(i) If the rotation of root y in T_0 is an unbalancing rotation, then, if it exists, the rotation of root y in T_1 is still an unbalancing rotation;*
- *(ii) If y has no left child in T_0, then y has no left child in T_1.*

Proof. *(i)*: If the rotation of root y in T_0 is an unbalancing rotation, it is because the imbalance values of y and its left child x do not satisfy (R1) or (R2) of Section 3.1. Thus, to change these imbalance values, one has to perform rotations to change the height of some subtrees of x and y. By Proposition 3.1, these rotations necessarily unbalance the obtained binary tree. Moreover, by Theorem 3.13, it is impossible to make rotations to balance it again. This shows that if y has a left child in T_1, it is necessarily a root of an unbalancing rotation.

(ii): This is immediate from the definition of the rotation operation and by the fact that the rotation operation does not change the infix order of the nodes of a binary tree. □

Lemma 6.1 shows that for all balanced binary trees T_0 and T_1 such that $T_0 \leq_{\mathrm{T}} T_1$, a node y cannot become a root of a conservative balancing rotation in T_1 if it is not a root of a conservative balancing rotation in T_0.

Lemma 6.2. *Let T_0 and T_1 be two balanced binary trees and y be a node of T_0 such that T_1 is obtained from T_0 by a rotation of root y. Then, denoting by x the left child of y in T_0, for all balanced binary tree T_2 such that $T_1 \leq_{\mathrm{T}} T_2$, x and y cannot be roots of conservative balancing rotations in T_2.*

Proof. Since T_1 is obtained by performing a conservative balancing rotation of root y into T_0, we have two cases to consider, following the imbalance values of x and y in T_0. If $\mathrm{i}_{T_0}(x) = \mathrm{i}_{T_0}(y) = -1$, then $\mathrm{i}_{T_1}(x) = \mathrm{i}_{T_1}(y) = 1$ and x and y are not roots of conservative balancing rotations in T_1, so that, by Lemma 6.1, x and y cannot be roots of conservative balancing rotations in T_2. If $\mathrm{i}_{T_0}(x) = 0$ and $\mathrm{i}_{T_0}(y) - 1$, then $\mathrm{i}_{T_1}(x) = 1$ and $\mathrm{i}_{T_1}(y) = 0$. For the same reason, x and y cannot be roots of conservative balancing rotations in T_2. □

A hypercube of dimension k can be seen as a poset whose elements are subsets of a set $\{e_1, \dots, e_k\}$, and ordered by the relation of inclusion. Let us denote by $\mathbb{H}_k$ the hypercube poset of dimension k.

We have the following characterization of the shape of balanced binary tree intervals:

Theorem 6.3. *Let T_0 and T_1 be two balanced binary trees such that $T_0 \leq_{\mathrm{T}} T_1$. Then, the poset $([T_0, T_1], \leq_{\mathrm{T}})$ is isomorphic to the hypercube $\mathbb{H}_k$, where k is the number of rotations needed to transform T_0 into T_1.*

Proof. First, note by Theorem 3.13, that the interval $I := [T_0, T_1]$ only contains balanced binary trees. Hence, all covering relations in I are conservative balancing rotations.

Denote by R the set of nodes y of T_0 such that y is a root of a rotation needed to transform T_0 into T_1. By Lemma 6.2, R is well defined — it is not a multiset — and if $y \in R$ then, denoting by x the left child of y in T_0, we have $x \notin R$. That implies that T_1 can be obtained from T_0 by performing, for all $y \in R$, a rotation of root y, independently of the order.

Let us now define a bijection between the elements of I and the set of the subsets of R. Let $T \in I$. By definition, it is possible to obtain T by performing some rotations from T_0. Let R_0 be the set of nodes which are roots of these rotations. Besides, it is possible to obtain T_1 by performing some rotations from T. Let R_1 be the set of nodes which are roots of these rotations. By Lemma 6.1, we have $R = R_0 \uplus R_1$ and thus $R_0 \subset R$. The set R_0 characterizes T. Conversely, for each subset $R_0 \subseteq R$ we can construct a unique binary tree $T \in I$. Indeed, T is obtained by doing the rotations of root y for all $y \in R_0$ into T_0, in any order. This is well-defined, by definition of R.

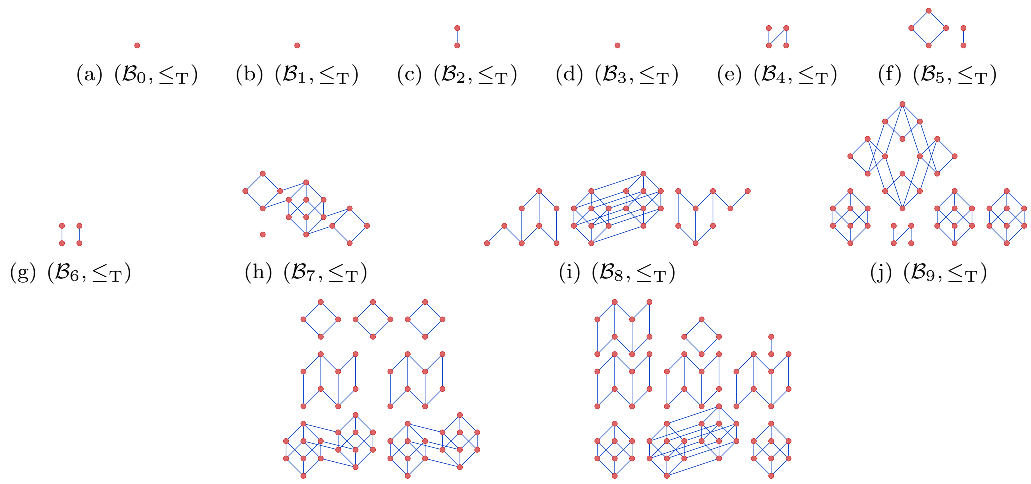
This shows that the interval I is isomorphic to the poset $\mathbb{H}_k$ where k is the number of rotations needed to transform T_0 into T_1. □

The first subposets of the Tamari lattice of balanced binary trees are depicted in Figure 14.

6.2. **Enumeration of balanced binary tree intervals.** Let us make use again of the synchronous grammars to enumerate balanced binary trees intervals.

Proposition 6.4. *The generating series enumerating balanced binary tree intervals in the Tamari lattice according to the number of leaves of the trees is $\mathcal{S}_{\mathrm{bi}}(x, 0, 0)$ where*

$$\mathcal{S}_{\mathrm{bi}}(x, y, z) = x + \mathcal{S}_{\mathrm{bi}}(x^2 + 2xy + yz, x, x^2 + xy). \tag{6.1}$$



(a) $(\mathcal{B}_0, \leq_{\mathrm{T}})$ (b) $(\mathcal{B}_1, \leq_{\mathrm{T}})$ (c) $(\mathcal{B}_2, \leq_{\mathrm{T}})$ (d) $(\mathcal{B}_3, \leq_{\mathrm{T}})$ (e) $(\mathcal{B}_4, \leq_{\mathrm{T}})$ (f) $(\mathcal{B}_5, \leq_{\mathrm{T}})$

(g) $(\mathcal{B}_6, \leq_{\mathrm{T}})$ (h) $(\mathcal{B}_7, \leq_{\mathrm{T}})$ (i) $(\mathcal{B}_8, \leq_{\mathrm{T}})$ (j) $(\mathcal{B}_9, \leq_{\mathrm{T}})$

(k) $(\mathcal{B}_{10}, \leq_{\mathrm{T}})$ (l) $(\mathcal{B}_{11}, \leq_{\mathrm{T}})$

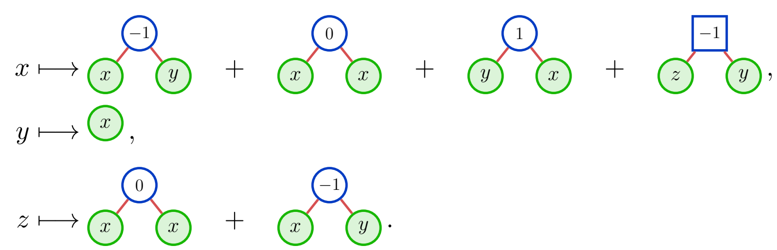
FIGURE 14. Hasse diagrams of the first $(\mathcal{B}_n, \leq_{\mathrm{T}})$ posets.

Proof. Let $I := [T_0, T_1]$ be a balanced binary tree interval and R be the set of nodes defined as in the proof of Theorem 6.3 associated with I. The proof of this theorem also shows that I can be encoded by T_0 in which the nodes of R are marked. To generate these objects, we consider a synchronous grammar which generates bud trees where (non-marked) nodes are and marked nodes are . Let us consider the synchronous grammar $S_{\mathrm{bi}} := \left(\{x, y, z\}, x, L\right)$ where L contains the substitution rules

(6.2)

(6.3)

(6.4)



We can apply the same idea developed in the proof of Proposition 4.12 to show that the bud trees generated by S_{bi} that only contain buds x have non-bud nodes labeled by their imbalance values. Hence, identifying in such trees the x with leaves, S_{bi} only generates balanced binary trees such that each of its node r with -1 as imbalance value can be marked provided that its left child has -1 or 0 as imbalance value and is not marked (recall that in this way, r is a root of a conservative balancing rotation). Indeed, if a x is substituted by a marked node, this marked node has a bud z as left child and z can only be substituted by a non-marked node with -1 or 0 as imbalance value. The role of the bud y is to delay the growth of a branch of the generated bud tree in order to create the imbalance values -1 and 1.

By setting $y \leq_B z \leq_B x$, S_{bi} satisfies the hypothesis of Lemma 4.6, and hence, is strict. This synchronous grammar also is unambiguous since it satisfies the hypothesis of Lemma 4.8. Indeed, the roots of the bud trees arising in a right member of the substitution rules of L are pairwise different, due to their labeling and their marking.

Finally, since S_{bi} is strict and unambiguous, by Proposition 4.11, we obtain the stated fixed-point functional equation, and the generating series is obtained by the specialization $y = 0$ and $z = 0$ in order to ignore bud trees that contain a bud labelled by y or by z. □

The solution of this fixed-point functional equation gives us the following first values for the number of balanced binary tree intervals in the Tamari lattice:

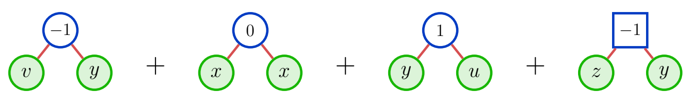
$$1, 1, 3, 1, 7, 12, 6, 52, 119, 137, 195, 231, 1019, 3503, 6593, 12616, 26178, 43500, 64157, 94688,\\ 232560, 817757, 2233757, 5179734, 11676838, 24867480. \tag{6.5}$$

The interval $[T_0, T_1]$ is a *maximal balanced binary tree interval* if T_0 (resp. T_1) is a minimal (resp. maximal) balanced binary tree.

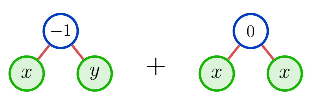
Proposition 6.5. *The generating series enumerating maximal balanced binary tree intervals in the Tamari lattice according to the number of leaves of the trees is* $\mathcal{S}_{\mathrm{mbi}}(x, 0, 0, 0)$ *where*

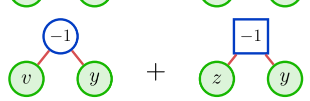
$$\mathcal{S}_{\mathrm{mbi}}(x, y, z, t) = x + \mathcal{S}_{\mathrm{mbi}}(x^2 + 2yt + yz, x, x^2 + xy, yt + yz). \tag{6.6}$$

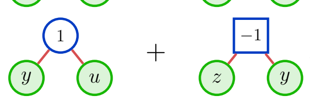
Proof. Let $I := [T_0, T_1]$ be a maximal balanced binary tree interval. This interval can be encoded by the minimal balanced binary tree T_0 in which the nodes that are roots of the conservative balancing rotations needed to transform T_0 into T_1 are marked. Moreover, since T_1 is a maximal balanced binary tree, by Proposition 5.3, it avoids the patterns of $P_{\max}$. Hence, the tree-like structure that encodes I must avoid the patterns of $P_{\min}$ and not have a node which is root of a conservative balancing rotation not marked if its parent or its left child is not marked. To generate these objects, we use the synchronous grammar $S_{\mathrm{mbi}} := \left(\{x, y, z, u, v\}, x, R\right)$ where R contains the substitution rules

(6.7) $x \longmapsto$ 

(6.8) $y \longmapsto x$,

(6.9) $z \longmapsto$ 

(6.10) $u \longmapsto$ 

(6.11) $v \longmapsto$ 

We can apply the same idea developed in the proof of Proposition 4.12 to show that the bud trees generated by S_{mbi} that only contain buds x have non-bud nodes labeled by their imbalance values. Hence, identifying in such trees the x with leaves, S_{mbi} only generates minimal balanced binary trees that are maximally marked. Indeed, by Proposition 5.3, the generated tree-like structures must avoid the two patterns of $P_{\min}$. To do that, we have to control the growth of the x when they are substituted by bud trees D whose roots are not marked and have an imbalance value of 1. Indeed, if the root of the right subtree of D grows with an imbalance value of 1 or 0, one of the two patterns is not avoided. The idea is to force the imbalance value of the root of the right subtree of D to be -1, role played by the bud u. Moreover, if the x are substituted by non-marked nodes a labeled by -1, to generate trees that are maximally marked, the left child of a has to be marked, or labeled by 1 (in this case, a is not root of a conservative balancing rotation). This is the role played by the bud v. The bud z appears in these substitution rules only as a left child of a marked node and it is substituted only by nodes with -1 or 0 as imbalance value, that are the only ones authorized for a left child of a root of a conservative balancing rotation. As usual, the role of the bud y is to delay the growth of a branch of the generated bud tree in order to create the imbalance values -1 and 1.

By setting $y \leq_B v \leq_B u \leq_B z \leq_B x$, S_{mbi} satisfies the hypothesis of Lemma 4.6, and hence, is strict. This synchronous grammar also is unambiguous since it satisfies the hypothesis of Lemma 4.8. Indeed, the roots of each bud trees arising in a right member of the substitution rules of R are different to one other, due to their labeling and their marking.

By Proposition 4.11, the fixed-point functional equation F associated with $\mathcal{S}_{\mathrm{mbi}}$ is

$$F(x,y,z,u,v) = x + F(x^2+yu+yv+yz, x, x^2+xy, yv+yz, yu+yz), \tag{6.12}$$

and, since the variables u and v play the same role, we obtain the stated fixed-point functional equation. The generating series is obtained by the specialization $y=0$, $z=0$ and $t=0$ in order to ignore bud trees that contain a bud labelled by y, z, u, or by v. □

The solution of this fixed-point functional equation gives us the following first values for the number of maximal balanced binary tree intervals in the Tamari lattice:

$$\begin{aligned}&1,1,1,1,3,2,2,6,9,15,15,17,41,77,125,178,252,376,531,740,1192,2179,4273,7738,\\&13012,20776,32389,49841,75457,113011,168888,252881,379348.\end{aligned} \tag{6.13}$$

We can slightly modify S_{mbi} to take into consideration the dimensions of the hypercubes isomorphic to the enumerated maximal balanced binary tree intervals. For that, we have to count the number of applications of substitution rules that generate a marked node. Let us use for that a parameter ξ. Whence we obtain the generating series defined by the fixed-point functional equation

$$\mathcal{S}_{\mathrm{mbi}}(x,y,z,t,\xi) = x + \mathcal{S}_{\mathrm{mbi}}(x^2+2yt+yz\xi, x, x^2+xy, yt+yz\xi, \xi). \tag{6.14}$$

First coefficients of x^i in $P := \mathcal{S}_{\mathrm{mbi}}(x,0,0,0,\xi)$ are

$$[x^1]\,P = 1, \tag{6.15}$$
$$[x^2]\,P = 1, \tag{6.16}$$
$$[x^3]\,P = \xi, \tag{6.17}$$
$$[x^4]\,P = 1, \tag{6.18}$$
$$[x^5]\,P = 3\xi, \tag{6.19}$$
$$[x^6]\,P = \xi+\xi^2, \tag{6.20}$$
$$[x^7]\,P = 2\xi, \tag{6.21}$$
$$[x^8]\,P = 1+4\xi^2+\xi^3, \tag{6.22}$$
$$[x^9]\,P = 4\xi+4\xi^2+\xi^4, \tag{6.23}$$
$$[x^{10}]\,P = 3\xi+9\xi^2+3\xi^3, \tag{6.24}$$
$$[x^{11}]\,P = 9\xi^2+6\xi^3, \tag{6.25}$$
$$[x^{12}]\,P = \xi+13\xi^2+2\xi^3+\xi^4, \tag{6.26}$$
$$[x^{13}]\,P = 6\xi+4\xi^2+16\xi^3+15\xi^4, \tag{6.27}$$
$$[x^{14}]\,P = 2\xi+18\xi^2+31\xi^3+12\xi^4+14\xi^5. \tag{6.28}$$

As example, the coefficient of x^{12} of $\mathcal{S}_{\mathrm{mbi}}(x,0,0,0,\xi)$ says that in the poset $(\mathcal{B}_{11}, \leq_{\mathrm{T}})$, there is one maximal 1-dimensional hypercube, thirteen maximal 2-dimensional hypercubes, two maximal 3-dimensional hypercubes and one maximal 4-dimensional hypercube (see Figure 14).

Note that Proposition 3.1 implies that all binary trees of the connected components of the posets $(\mathcal{B}_n, \leq_{\mathrm{T}})$ have same height. However, the converse is false: There is two connected components in the poset $(\mathcal{B}_5, \leq_{\mathrm{T}})$ and each binary tree of $\mathcal{B}_5$ has same height.

7. Intervals of other binary trees families in the Tamari lattice

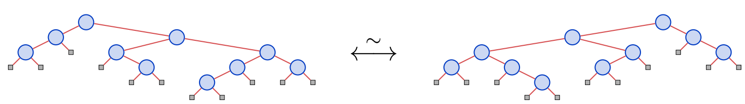
7.1. Generalized balanced binary trees.

7.1.1. *Definitions.* Let V be a subset of $\mathbb{Z}$. We say that a binary tree T is V*-balanced* if for all node x of T, $\mathrm{i}_T(x) \in V$. Let us denote by $\mathcal{B}^V$ the set of V-balanced binary trees. Note that the set of balanced binary trees is $\mathcal{B}^{[-1,1]}$. It is clear that 0 must always belongs to V since a binary tree necessarily has a node with both empty left and right subtrees; Otherwise, $\mathcal{B}^V$ would be empty. A natural question about V-balanced binary trees demands to characterize the sets V such that $\mathcal{B}^V$ is closed by interval in the Tamari lattice.

Let T be a binary tree. Denote by $T^\sim$ the binary tree obtained by exchanging the right and left subtrees of each of its nodes. More formally,

$$T^\sim := \begin{cases} R^\sim \wedge L^\sim & \text{if } T = L \wedge R, \\ \perp & \text{otherwise } (T = \perp). \end{cases} \tag{7.1}$$

For instance, one has

(7.2) 

If V is a subset of $\mathbb{Z}$, let us also denote by $V^\sim$ the set $\{-v : v \in V\}$.

7.1.2. *A symmetry.*

Lemma 7.1. *Let T_0 and T_1 be two binary trees such that $T_0 \leq_{\mathrm{T}} T_1$. Then, $T_1^\sim \leq_{\mathrm{T}} T_0^\sim$.*

Proof. Assume that $S_0 \lessdot S_1$ where $S_0 = (A \wedge B) \wedge C$ and $S_1 = A \wedge (B \wedge C)$. Hence, we have $S_1^\sim = (C^\sim \wedge B^\sim) \wedge A^\sim$ and $S_0^\sim = C^\sim \wedge (B^\sim \wedge A^\sim)$. Thus, $S_1^\sim \lessdot S_0^\sim$, and the result follows from the fact that $\leq_{\mathrm{T}}$ is the reflexive and transitive closure of $\lessdot$. □

Lemma 7.2. *For all $V \subseteq \mathbb{Z}$, the application $\sim$ yields a bijection between the sets $\mathcal{B}^V$ and $\mathcal{B}^{V^\sim}$.*

Proof. It is immediate, from the definition of $\sim$, that the application $\sim$ is an involution. It then remains to show that if $T \in \mathcal{B}^V$, then $T^\sim \in \mathcal{B}^{V^\sim}$. Let x be a node of T and L (resp. R) be the left (resp. right) subtree of x. We have $v := \mathrm{i}_T(x) = \mathrm{h}(R) - \mathrm{h}(L) \in V$. In $T^\sim$, one has $\mathrm{i}_{T^\sim}(x) = \mathrm{h}(L^\sim) - \mathrm{h}(R^\sim) = \mathrm{h}(L) - \mathrm{h}(R) = -v \in V^\sim$. Hence, $T^\sim \in \mathcal{B}^{V^\sim}$. □

Proposition 7.3. *For all $V \subseteq \mathbb{Z}$, the set $\mathcal{B}^V$ is closed by interval in the Tamari lattice if and only if the set $\mathcal{B}^{V^\sim}$ also is.*

Proof. Assume that $\mathcal{B}^{V^\sim}$ is closed by interval in the Tamari lattice. By contradiction, assume that there exist $T_0, T_2 \in \mathcal{B}^V$ and $T_1 \notin \mathcal{B}^V$ such that $T_0 \leq_{\mathrm{T}} T_1 \leq_{\mathrm{T}} T_2$. By Lemma 7.1, we have $T_2^\sim \leq_{\mathrm{T}} T_1^\sim \leq_{\mathrm{T}} T_0^\sim$, and by Lemma 7.2, $T_0^\sim, T_2^\sim \in \mathcal{B}^{V^\sim}$ and $T_1^\sim \notin \mathcal{B}^{V^\sim}$. That implies that $\mathcal{B}^{V^\sim}$ is not closed by interval in the Tamari lattice, which is contradictory with our hypothesis. □

7.1.3. $\{0,1\}$-*balanced binary trees.* Using the methods developed in Section 4, one can enumerate $\{0,1\}$-balanced binary trees according to their number of leaves, and obtain the fixed-point functional equation

$$\mathcal{S}_{01}(x,y) = x + \mathcal{S}_{01}(x^2 + xy, x), \tag{7.3}$$

where the generating series of $\{0,1\}$-balanced binary trees is the specialization $\mathcal{S}_{01}(x,0)$. First values are

$$1, 1, 1, 1, 1, 2, 2, 2, 3, 5, 7, 9, 11, 13, 17, 26, 42, 66, 97, 134, 180, 241, 321, 424, 564, 774, 1111. \tag{7.4}$$

Proposition 7.4. *The set of $\{0,1\}$-balanced binary trees is closed by interval in The Tamari lattice.*

Proof. Let $T_0 \in \mathcal{B}^{\{0,1\}}$. Since T_0 is only composed of nodes with 0 or 1 as imbalance value, one can only perform into T_0 rotations of the kind (R3), (R5), (R8) or (R9) studied in Section 3.1. Since these rotations are unbalancing rotations, for all binary tree T_1 such that $T_0 \lessdot T_1$, T_1 is not balanced and hence, $T_1 \notin \mathcal{B}^{\{0,1\}}$. By Theorem 3.13, for all binary tree T_2 such that $T_1 \leq_{\mathrm{T}} T_2$, T_2 is not balanced, and with greater reason, $T_2 \notin \mathcal{B}^{\{0,1\}}$. Therefore, $\mathcal{B}^{\{0,1\}}$ is closed by interval in the Tamari lattice. □

The proof of Proposition 7.4 also shows that every rotation performed into a $\{0,1\}$-balanced binary tree gives a $\{0,1\}$-unbalanced binary tree. That implies that any pair of elements of $\mathcal{B}^{\{0,1\}}$ is incomparable.

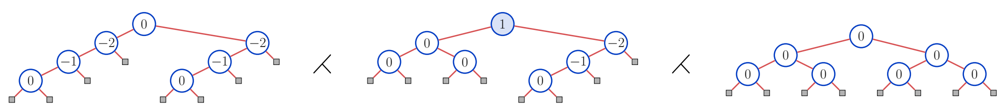
Computer trials suggest that for all $\beta \in \mathbb{Z}$, any pair of elements of $\mathcal{B}^{\{0,\beta\}}$ is incomparable. Hence, the sets $\mathcal{B}^{\{0,\beta\}}$ seem to be closed by interval in the Tamari lattice.

7.1.4. $[-\alpha,\beta]$-*balanced binary trees.*

Lemma 7.5. *For all $\alpha \geq 2$, the sets $\mathcal{B}^{[-\alpha,0]}$ and $\mathcal{B}^{]-\infty,0]}$ are not closed by interval in the Tamari lattice.*

Proof. It is enough to exhibit a chain of the sort $T_0 \lessdot T_1 \lessdot T_2$ where $T_0, T_2 \in \mathcal{B}^{[-\alpha,0]} \cap \mathcal{B}^{]-\infty,0]}$ and $T_1 \notin \mathcal{B}^{[-\alpha,0]} \cup \mathcal{B}^{]-\infty,0]}$. The following chain, where nodes are labeled by their imbalance values, is the case:

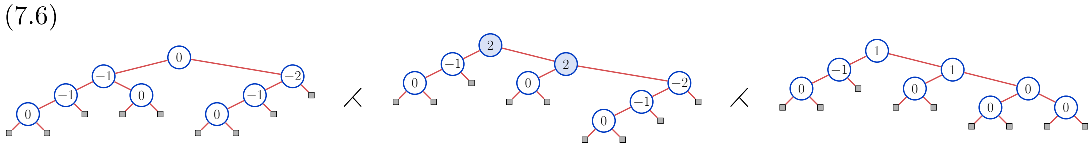
(7.5)



□

Lemma 7.6. *For all $\alpha \geq 2$, the sets $\mathcal{B}^{[-\alpha,1]}$ and $\mathcal{B}^{]-\infty,1]}$ are not closed by interval in the Tamari lattice.*

Proof. It is enough to exhibit a chain of the sort $T_0 \lessdot T_1 \lessdot T_2$ where $T_0, T_2 \in \mathcal{B}^{[-\alpha,1]} \cap \mathcal{B}^{]-\infty,1]}$ and $T_1 \notin \mathcal{B}^{[-\alpha,1]} \cup \mathcal{B}^{]-\infty,1]}$. The following chain, where nodes are labeled by their imbalance values, is the case:
(7.6)

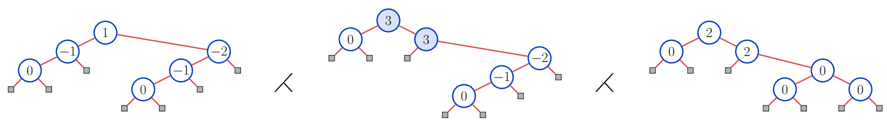


□

Lemma 7.7. *For all $\alpha \geq 2$, the sets $\mathcal{B}^{[-\alpha,2]}$ and $\mathcal{B}^{]-\infty,2]}$ are not closed by interval in the Tamari lattice.*

Proof. It is enough to exhibit a chain of the sort $T_0 \lessdot T_1 \lessdot T_2$ where $T_0, T_2 \in \mathcal{B}^{[-\alpha,2]} \cap \mathcal{B}^{]-\infty,2]}$ and $T_1 \notin \mathcal{B}^{[-\alpha,2]} \cup \mathcal{B}^{]-\infty,2]}$. The following chain, where nodes are labeled by their imbalance values, is the case:

(7.7)

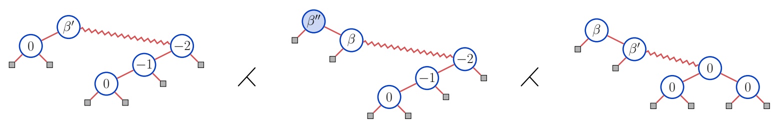


□

Lemma 7.8. *For all $\alpha \geq 2$ and $\beta \geq 3$, the sets $\mathcal{B}^{[-\alpha,\beta]}$ and $\mathcal{B}^{]-\infty,\beta]}$ are not closed by interval in the Tamari lattice.*

Proof. It is enough to exhibit a chain of the sort $T_0 \lessdot T_1 \lessdot T_2$ where $T_0, T_2 \in \mathcal{B}^{[-\alpha,\beta]} \cap \mathcal{B}^{]-\infty,\beta]}$ and $T_1 \notin \mathcal{B}^{[-\alpha,\beta]} \cup \mathcal{B}^{]-\infty,\beta]}$. By setting $\beta' := \beta - 1$ and $\beta'' := \beta + 1$, the following generic chain, where nodes are labeled by their imbalance values, and where the edges depicted by ⏦ denote a right comb binary tree with $\beta - 3$ nodes, is the case:

(7.8)



□

Theorem 7.9. *Let V be an interval of $\mathbb{Z}$ containing 0. The set $\mathcal{B}^V$ is closed by interval in the Tamari lattice if and only if $V \in \{\{0\}, \{-1,0\}, \{0,1\}, \{-1,0,1\}, \mathbb{Z}\}$.*

Proof. Since $\mathcal{B}^{\{0\}}$ only contains perfect binary trees and there is at most one such element with a given number of nodes, $\mathcal{B}^{\{0\}}$ is closed by interval. Moreover, by Proposition 7.4, $\mathcal{B}^{\{0,1\}}$ is closed by interval, and by Proposition 7.3, $\mathcal{B}^{\{-1,0\}}$ also is. By Theorem 3.13, $\mathcal{B}^{\{-1,0,1\}}$ is closed by interval. Finally, since $\mathcal{B}^{\mathbb{Z}} = \mathcal{T}$, $\mathcal{B}^{\mathbb{Z}}$ is obviously closed by interval.

If V is an interval of $\mathbb{Z}$ containing 0 and that does not fit into the previous cases, necessarily V or $V^{\sim}$ satisfies the assumptions of Lemma 7.5, 7.6, 7.7, or 7.8. Thus, by Proposition 7.3, $\mathcal{B}^V$ is not closed by interval. □

Theorem 7.9 emphasizes the special role played by balanced binary trees in the Tamari lattice. Indeed, the interval $V := [-1,1]$ of $\mathbb{Z}$ is the only interval different from $\mathbb{Z}$ such that $\mathcal{B}^V$ is closed by interval in the Tamari lattice and such that the subposet of the Tamari lattice induced by $\mathcal{B}^V$ contains nontrivial intervals (see Theorem 6.3 and Figure 14).

7.2. **Weight balanced binary trees.** Denote by $\mathrm{n}(T)$ the number of nodes of the binary tree T. Let us define the *weight imbalance mapping* wi_T which associates an element of $\mathbb{Z}$ with a node x of T. It is defined by

$$\mathrm{wi}_T(x) := \mathrm{n}(R) - \mathrm{n}(L), \tag{7.9}$$

where L (resp. R) is the left (resp. right) subtree of x. A node x is *weight balanced* if

$$\mathrm{wi}_T(x) \in \{-1, 0, 1\}. \tag{7.10}$$

Definition 7.10. *A binary tree T is* weight balanced *if all nodes of T are weight balanced.*

The sequence $(w_n)_{n\geq 0}$ of the number of weight balanced binary trees with n nodes satisfies straightforwardly the recurrence relation

$$w_n = \begin{cases} 1 & \text{if } n \in \{0,1\}, \\ 2w_k w_{k-1} & \text{if } n = 2k, \\ w_k^2 & \text{where } n = 2k+1, \text{ otherwise.} \end{cases} \tag{7.11}$$

This is Sequence **A110316** of [Slo]. First values are

$$1, 1, 2, 1, 4, 4, 4, 1, 8, 16, 32, 16, 32, 16, 8, 1, 16, 64, 256, 256, 1024, 1024. \tag{7.12}$$

Lemma 7.11. *For all nonempty weight balanced binary tree T, the following relation between its height and its number of nodes holds*

$$\mathrm{h}(T) = \lfloor \log_2(\mathrm{n}(T)) \rfloor + 1. \tag{7.13}$$

Proof. We proceed by structural induction on the set of nonempty weight balanced binary trees. The lemma is true for the one-node binary tree. Assume now that (7.13) holds for both the weight balanced binary trees L and R such that $T := L \wedge R$ is weight balanced. We have now two cases to consider, depending if L and R have the same number of nodes or not. If $\mathrm{n}(L) = \mathrm{n}(R)$, set $k := \mathrm{n}(L)$. We have

$$\lfloor \log_2(\mathrm{n}(T)) \rfloor + 1 = \lfloor \log_2(2k+1) \rfloor + 1 \tag{7.14}$$
$$= \lfloor \log_2(2) + \log_2(k + 1/2) \rfloor + 1 \tag{7.15}$$
$$= \lfloor \log_2(k + 1/2) \rfloor + 2 \tag{7.16}$$
$$= \lfloor \log_2(k) \rfloor + 2 \tag{7.17}$$
$$= \mathrm{h}(L) + 1 \tag{7.18}$$
$$= \mathrm{h}(R) + 1 \tag{7.19}$$
$$= \mathrm{h}(T). \tag{7.20}$$

The equality between (7.16) and (7.17) is provided by the fact that k is an integer. The equality between (7.17) and (7.18) follows by induction hypothesis.

If $\mathrm{n}(L) \neq \mathrm{n}(R)$, assume without lost of generality that $\mathrm{n}(L) = \mathrm{n}(R) + 1$ and set $k := \mathrm{n}(L)$. An analog computation as above implies (7.13). □

Proposition 7.12. *The set of weight balanced binary trees is a subset of the set of the (height) balanced binary trees.*

Proof. We proceed by structural induction on the set of weight balanced binary trees to show that each weight balanced binary tree is also (height) balanced. This property is true for both the empty tree and the one-node binary tree. Assume now that this property holds for two weight balanced binary trees L and R such that $T := L \wedge R$ is weight balanced. By Lemma 7.11, we have

$$\mathrm{h}(R) - \mathrm{h}(L) = \lfloor \log_2(\mathrm{n}(R)) \rfloor - \lfloor \log_2(\mathrm{n}(L)) \rfloor, \tag{7.21}$$

and since T is weight balanced, we have $|\,\mathrm{n}(R) - \mathrm{n}(L)| \leq 1$ so that $|\,\mathrm{h}(R) - \mathrm{h}(L)| \leq 1$. By induction hypothesis, L and R are (height) balanced, proving that T also is. □

Proposition 7.13. *Let T_0 and T_1 be two weight balanced binary trees such that $T_0 \leq_{\mathrm{T}} T_1$. Then, the interval $[T_0, T_1]$ only contains weight balanced binary trees.*

Proof. Let us show that for all binary tree T, any rotation operation performed into T does not decrease the weight imbalance values of any node of T. Let y be a node in T and x its left child. Let $(A \wedge B) \wedge C$ be the subtree of root y in T. Let T' be the binary tree obtained by the rotation of root y from T. We have the following weight imbalance values:

$$\begin{cases} \mathrm{wi}_T(x) = \mathrm{n}(B) - \mathrm{n}(A), \\ \mathrm{wi}_T(y) = \mathrm{n}(C) - \mathrm{n}(B) - \mathrm{n}(A) - 1, \end{cases} \tag{7.22}$$

and

$$\begin{cases} \mathrm{wi}_{T'}(x) = \mathrm{n}(B) + \mathrm{n}(C) + 1 - \mathrm{n}(A), \\ \mathrm{wi}_{T'}(y) = \mathrm{n}(C) - \mathrm{n}(B), \end{cases} \tag{7.23}$$

showing that $\mathrm{wi}_{T'}(x) > \mathrm{wi}_T(x)$ and $\mathrm{wi}_{T'}(y) > \mathrm{wi}_T(y)$. Besides, note that the rotation does not change the weight imbalance values of the other nodes of T.

This shows that the set of weight balanced binary trees is closed by interval in the Tamari lattice since, by starting from a weight balanced binary tree T and by performing a rotation that gives a weight unbalanced binary tree T', there exists a node z of T' such that $\mathrm{wi}_{T'}(z) \geq 2$ and it is impossible to decrease this value so that each binary tree greater than T' is not weight balanced. □

Note that the proof of Proposition 7.13 also proves that for all $k \geq 0$, the sets of *k-weight balanced binary trees*, that are the sets of binary trees T such that for all node x of T, the relation $|\mathrm{wi}_T(x)| \leq k$ holds, are closed by interval in the Tamari lattice.

Since by Proposition 7.12, weight balanced binary trees are also (height) balanced, by Proposition 7.13 and Theorem 6.3, the intervals of weight balanced binary trees are isomorphic to a hypercube. However, the set of weight balanced binary trees has an additional property:

Proposition 7.14. *The restriction of the Tamari order on the set of weight balanced binary trees is a graded poset.*

Proof. Let us characterize the conservative weight balancing rotations. Let $T_0 := (A \wedge B) \wedge C$ and $T_1 := A \wedge (B \wedge C)$ be two weight balanced binary trees such that T_1 is obtained by a rotation at the root y of T_0. Denote by x the left child of y in T_0. Note that the rotation that transforms T_0 into T_1 cannot be a conservative weight balancing rotation if $\mathrm{wi}_{T_0}(x) = 1$ or $\mathrm{wi}_{T_0}(y) = 1$ since, following the proof of Proposition 7.13, the imbalance values of x and y both increase after a rotation. Here follows the list of the weight imbalance values of the nodes x and y in T_0 and T_1 expressed as $(\mathrm{wi}_{T_0}(x), \mathrm{wi}_{T_0}(y)) \longrightarrow (\mathrm{wi}_{T_1}(x), \mathrm{wi}_{T_1}(y))$:

(R'1) $(-1,-1) \longrightarrow (2\,\mathrm{n}(A) - 1, \mathrm{n}(A))$,

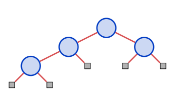
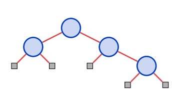
(R'2) $(0,-1) \longrightarrow (2\,\mathrm{n}(A) + 1, \mathrm{n}(A))$,

(R'3) $(-1,0) \longrightarrow (2\,\mathrm{n}(A), \mathrm{n}(A) + 1)$,

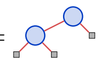
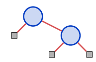
(R'4) $(0,0) \longrightarrow (2\,\mathrm{n}(A) + 2, \mathrm{n}(A) + 1)$.

Hence, we have four kind of rotations to explore:

Case 1: Regarding (R'1), we necessarily have $\mathrm{n}(A) = 1$. Indeed, if $\mathrm{n}(A) \geq 2$, y would not be weight balanced in T_1, and if $\mathrm{n}(A) = 0$, since $\mathrm{wi}_{T_0}(x) = -1$, that would imply that $\mathrm{n}(B) = -1$, which is absurd. Hence, since $\mathrm{n}(A) = 1$, we have $\mathrm{n}(B) = 0$ and $\mathrm{n}(C) = 1$. Thus, there is only one pair (T_0, T_1) satisfying this kind of conservative weight balancing rotation:

(7.24) $T_0 =$  $\longrightarrow$  $= T_1.$

Case 2: Concerning (R'2), we necessarily have $\mathrm{n}(A) = 0$. Indeed, if $\mathrm{n}(A) \geq 1$, x would not be weight balanced in T_1. Hence, since $\mathrm{n}(A) = 0$, we have $\mathrm{n}(B) = 0$ and $\mathrm{n}(C) = 0$. Thus, there is only one pair (S_0, S_1) that satisfies this kind of conservative weight balancing rotation:

(7.25) $S_0 =$  $\longrightarrow$  $= S_1.$

Case 3: Regarding (R'3), we necessarily have $\mathrm{n}(A) = 0$. That implies $\mathrm{h}(B) = -1$, which is absurd. Hence, (R'3) cannot be a conservative weight balancing rotation.
Case 4: Concerning (R'4), x satisfies $\mathrm{wi}_{T_1}(x) \geq 2$, and thus (R'4) is not a case of a conservative weight balancing rotation.

Hence, we only have two sorts of conservative weight balancing rotations. They are the ones depicted in (7.24) and (7.25).

Since each such rotation suppresses exactly one subtree of the form S_0 and adds exactly one subtree of the form S_1, we can define a map $\phi : \mathcal{T} \to \mathbb{N}$ where $\phi(T)$ is the number of subtrees of the form S_1 in T. Hence, since by Proposition 7.13 the covering relations of the Tamari lattice restricted to the weight balanced binary trees are only conservative weight balancing rotations, the statistic ϕ is a ranking function of the Tamari lattice restricted to these elements, and shows that this poset is graded. □

7.3. **Binary trees with fixed canopy.** The *canopy* $\mathrm{cnp}(T)$ (see [LR98] and [Vie04]) of a binary tree T is the word on the alphabet $\{0, 1\}$ obtained by browsing the leaves of T from left to right except the first and the last one, writing 0 if the considered leaf is oriented to the right, 1 otherwise (see Figure 15).

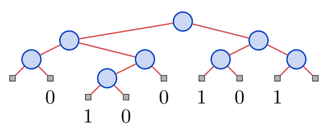


FIGURE 15. The canopy of this binary tree is 0100101.

For all $u \in \{0, 1\}^*$, define the set $\mathcal{C}_u$ by

$$\mathcal{C}_u := \{T \in \mathcal{T} : \mathrm{cnp}(T) = u\}\,. \tag{7.26}$$

Note that the sets of binary trees with a given canopy play a role in a injective Hopf morphism relating the Hopf algebra of noncommutative symmetric functions **Sym** [GKL+94] and the Hopf algebra of binary trees **PBT** [LR98, HNT05]. Recall that the fundamental basis of **PBT** is $\{\mathbf{P}_T\}_{T \in \mathcal{T}}$ and is indexed by binary trees. One can see the fundamental basis of **Sym** as a basis $\{\mathbf{P}_u\}_{u \in \{0,1\}^*}$ indexed by binary words. The injective Hopf morphism $\beta : \mathbf{Sym} \hookrightarrow \mathbf{PBT}$ also satisfies (see [Gir11])

$$\beta(\mathbf{P}_u) = \sum_{T \,\in\, \mathcal{C}_u} \mathbf{P}_T. \tag{7.27}$$

Proposition 7.15. *For all $u \in \{0, 1\}^*$, the set $\mathcal{C}_u$ is an interval of the Tamari lattice.*

Proof. Let us prove first that $\mathcal{C}_u$ is closed by interval in the Tamari lattice. Consider a binary tree T_0 and y one of its nodes. Let $(A \wedge B) \wedge C$ the subtree of T_0 of root y and T_1 be the binary tree obtained by the rotation of root y from T_0. Regardless A and C, if B is not empty, we have $\mathrm{cnp}(T_0) = \mathrm{cnp}(T_1)$; Otherwise, B is a leaf and its orientation changes from right to left. Thus, $\mathrm{cnp}(T_1)$ is lexicographically not smaller than $\mathrm{cnp}(T_0)$, which proves that $\mathcal{C}_u$ is closed by interval.

We give now a counting argument to prove that $\mathcal{C}_u$ also is an interval of the Tamari lattice. Let T be a maximal element among $\mathcal{C}_u$. Thus, each rotation changes the canopy of T, and hence, for every node y which has a left child x in T, x has no right child. The set of such maximal binary trees, denoted $\mathcal{M}$, is characterized by the following regular specification (see [FS09] for a general survey on regular specifications):

$$\mathcal{M} = \mathcal{L} \times \left\{ \bigcirc \right\} \times \mathcal{M} \quad + \quad \{\perp\}, \tag{7.28}$$

where $\mathcal{L}$ is the set of left comb binary trees. It admits the following generating series $M(x)$, which enumerates the elements of $\mathcal{M}$ according to their number of nodes:

$$M(x) = \frac{1-x}{1-2x} = 1 + \sum_{n \geq 1} 2^{n-1} x^n. \tag{7.29}$$

Moreover, for all $n \geq 1$ there are exactly 2^{n-1} sets $\mathcal{C}_u$ where $\ell(u) = n-1$, and there are the same number of such maximal binary trees. That implies that there is exactly one maximal element in each $\mathcal{C}_u$. By the same reasoning, we can show that there is exactly one minimal tree in each $\mathcal{C}_u$, proving the result. □

The statement of Proposition 7.15 is already known [LR02], only our proof is new.

7.4. **Narayana binary trees.** Let T be a binary tree. Denote by $\mathrm{nar}(T)$ the number of nodes of T that have a nonempty right child. We say that T is a *k-Narayana binary tree* if $\mathrm{nar}(T) = k$. These binary trees are enumerated by the Narayana numbers [Nar55] (see Sequence **A001263** of [Slo]). First values are

n	$\#\{T \in \mathcal{T}_n : \mathrm{nar}(T) = k\}$, $k = 0, \dots, n-1$							
1	1							
2	1	1						
3	1	3	1					
4	1	6	6	1				
5	1	10	20	10	1			
6	1	15	50	50	15	1		
7	1	21	105	175	105	21	1	
8	1	28	196	490	490	196	28	1

Proposition 7.16. *For all $k \geq 0$ and T_0 and T_1 two k-Narayana binary trees such that $T_0 \leq_{\mathrm{T}} T_1$, the interval $[T_0, T_1]$ only contains k-Narayana binary trees.*

Proof. Consider a node y of T_0 and let $(A \wedge B) \wedge C$ the subtree of T_0 of root y and T_1 be the binary tree obtained by the rotation of root y from T_0. Regardless A and C, if B is not empty, T_0 and T_1 have the same number of nodes that have a right child; Otherwise, the number of right children increases by one in T_1. Hence, in every chain $T_0 \leq_{\mathrm{T}} T_1 \leq_{\mathrm{T}} \dots \leq_{\mathrm{T}} T_\ell$, we have $\mathrm{nar}(T_0) \leq \mathrm{nar}(T_1) \leq \dots \leq \mathrm{nar}(T_\ell)$. That proves that the set of k-Narayana binary trees is closed by interval in the Tamari lattice. □

Proposition 7.17. *For all $k \geq 0$, the set of k-Narayana binary trees with n nodes is the disjoint union of the sets $\mathcal{C}_u$ where $\ell(u) = n-1$ and u contains k occurrences of* 1.

Proof. It is enough to show that for all binary tree T of canopy u, the number of 1 in u is $\mathrm{nar}(u)$. Let us show this property by structural induction on the set of binary trees. If T is empty, this property is clearly satisfied. Assume now that $T := L \wedge R$, and set $v := \mathrm{cnp}(L)$ and $w := \mathrm{cnp}(R)$. We have now to deal four cases whether L and R are empty or not.

Case 1: If L and R are empty, T is the one-node binary tree and the property is satisfied.

Case 2: If L and R are both not empty, then $\mathrm{cnp}(T) = v.0.1.w$. Since $\mathrm{nar}(T) = \mathrm{nar}(L) + \mathrm{nar}(R) + 1$, by induction hypothesis, the property is satisfied.
Case 3: If L is empty and R not, then $\mathrm{cnp}(T) = 1.w$. Since $\mathrm{nar}(T) = \mathrm{nar}(R) + 1$, by induction hypothesis, the property is satisfied.
Case 4: If R is empty and L not, then $\mathrm{cnp}(T) = v.0$. Since $\mathrm{nar}(T) = \mathrm{nar}(L)$, by induction hypothesis, the property is satisfied.

□

Corollary 7.18. *For all $k \geq 0$, the set of k-Narayana binary trees with n nodes is a disjoint union of intervals in the Tamari lattice.*

Proof. The property follows from the fact that the set of k-Narayana binary trees with n nodes is the union of some binary trees with a given canopy (Proposition 7.17) and that the sets of binary trees with a given canopy are intervals of the Tamari lattice (Proposition 7.15). □

References

[AU94] A. Aho and J. Ullman. *Foundations of Computer Science*. W. H. Freeman, 1994.
[AVL62] G.M. Adelson-Velsky and E. M. Landis. An algorithm for the organization of information. *Soviet Mathematics Doklady*, 3:1259–1263, 1962.
[BB09] O. Bernardi and N. Bonichon. Catalan's intervals and realizers of triangulations. *Journal of Combinatorial Theory Series A*, 116:55–75, 2009.
[BLL88] F. Bergeron, G. Labelle, and P. Leroux. Functional Equations for Data Structures. *Lecture Notes in Computer Science*, 294:73–80, 1988.
[BLL94] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial Species and Tree-like Structures*. Cambridge University Press, 1994.
[CLRS03] T.H. Cormen, C. E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to algorithms*. McGraw-Hill, 2003.
[FS09] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.
[Gir10] S. Giraudo. Balanced binary trees in the Tamari lattice. *Formal Power Series and Algebraic Combinatorics*, 22:596–607, 2010.
[Gir11] S. Giraudo. Algebraic and combinatorial structures on Baxter permutations. *Formal Power Series and Algebraic Combinatorics*, 23:387–398, 2011.
[GKL+94] I.M. Gelfand, D. Krob, A. Lascoux, B. Leclerc, V.S. Retakh, and J.-Y. Thibon. Noncommutative symmetric functions I. `arXiv:hep-th/9407124v1`, 1994.
[HNT05] F. Hivert, J.-C. Novelli, and J.-Y. Thibon. The Algebra of Binary Search Trees. *Theoretical Computer Science*, 339, Issue 1:129–165, 2005.
[HT72] S. Huang and D. Tamari. Problems of associativity: A simple proof for the lattice property of systems ordered by a semi-associative law. *Journal of Combinatorial Theory, Series A*, 13:7–13, 1972.
[Knu98] D. Knuth. *The Art of Computer Programming. Volume 3. Sorting and searching*. Addison Wesley Longman, 1998.
[Knu04] D. Knuth. *The Art of Computer Programming. Volume 4, Fascicle 4. Generating all trees - History of combinatorial generation*. Addison Wesley Longman, 2004.
[LR98] J.-L. Loday and M. O. Ronco. Hopf Algebra of the Planar Binary Trees. *Advances in Mathematics*, 139:293–309, 1998.
[LR02] J.-L. Loday and M. Ronco. Order Structure on the Algebra of Permutations and of Planar Binary Trees. *Journal of Algebraic Combinatorics*, Volume 15, Number3:253–270, 2002.
[Nar55] T.V. Narayana. Sur les treillis formés par les partitions d'un entier. *C.R. Acad. Sci. Paris Ser. I.*, 240:1188, 1955.
[Odl82] A. M. Odlyzko. Periodic Oscillations of Coefficients of Power Series That Satisfy Functional Equations. *Advances in Mathematics*, 44:180–205, 1982.
[S+10] W. A. Stein et al. *Sage Mathematics Software (Version 4.2)*. The Sage Development Team, 2010. http://www.sagemath.org.
[Slo] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences. http://www.research.att.com/~njas/sequences/.
[Sta90] R. P. Stanley. *Enumerative Combinatorics II*. Cambridge University Press, 1990.

[Tam62] D. Tamari. The algebra of bracketings and their enumeration. *Nieuw Arch. Wisk. (3)*, 10:131–146, 1962.

[Vie04] X. Viennot. *Up-down sequences of permutations, paths and canopy of binary trees*, 2004. invited talk at the Lascouxfest, 52nd SLC, Otrott.

Institut Gaspard-Monge, université Paris-Est Marne-la-Vallée, 5 Boulevard Descartes, Champs-sur-Marne, 77454, Marne-la-Vallée cedex 2, France

E-mail address: samuele.giraudo@univ-mlv.fr