On trees, tanglegrams, and tangled chains

Sara C Billey, Matjaz Konvalinka, Frderick A. Matsen Iv

FPSAC 2016 Vancouver, Canada DMTCS proc. **BC**, 2016, 263–274

On trees, tanglegrams, and tangled chains

Sara C. Billey[1†], Matjaž Konvalinka[2‡], and Frederick A. Matsen IV[3§]

[1]*Department of Mathematics, University of Washington, Seattle, WA 98195, USA*
[2]*Department of Mathematics, University of Ljubljana & Institute for Mathematics, Physics and Mechanics, Ljubljana, Slovenia*
[3]*Computational Biology Program, Fred Hutchinson Cancer Research Center,Seattle, WA 98109, USA*

Abstract. Tanglegrams are a class of graphs arising in computer science and in biological research on cospeciation and coevolution. They are formed by identifying the leaves of two rooted binary trees. The embedding of the trees in the plane is irrelevant for this application. We give an explicit formula to count the number of distinct binary rooted tanglegrams with n matched leaves, along with a simple asymptotic formula and an algorithm for choosing a tanglegram uniformly at random. The enumeration formula is then extended to count the number of tangled chains of binary trees of any length. This work gives a new formula for the number of binary trees with n leaves. Several open problems and conjectures are included along with pointers to several followup articles that have already appeared.

Résumé. Les tanglegrams sont une classe de graphes qui apparaissent en informatique et en biologie dans le contexte de la cospéciation et de la coévolution. Ils sont formés en identifiant les feuilles de deux arbres binaires enracinés (les deux arbres n'étant pas munis d'un plongement). Nous donnons une formule explicite pour compter le nombre de tanglegrams binaires enracinés distincts avec n feuilles appariées, ainsi qu'une formule asymptotique simple et un algorithme pour engendrer un tanglegram aléatoire de manière uniforme. La formule de dénombrement est ensuite étendue pour compter le nombre de chaînes enchevêtrées d'arbres binaires de longueur quelconque. Cette analyse donne une nouvelle formule pour le nombre d'arbres binaires (non plongés) avec n feuilles. Plusieurs problèmes ouverts et conjectures sont inclus ainsi que les références de plusieurs articles complémentaires qui ont déjà paru.

Keywords. tanglegrams, enumeration, binary trees, binary partitions

1 Introduction

Tanglegrams are graphs obtained by taking two binary rooted trees with the same number of leaves and matching each leaf from the tree on the left with a unique leaf from the tree on the right. The embedding of the trees in the plane is irrelevant for this application. This construction is used in the study of cospeciation and coevolution in biology. The embedding of the trees in the plane is irrelevant for this application. For example, the tree on the left may represent the phylogeny of a host, such as gopher, while the tree on the right may represent a parasite, such as louse [19, page 71]. One important problem is

†Partially supported by the National Science Foundation grant DMS-1101017.

‡Email: matjaz.konvalinka@fmf.uni-lj.si. Partially supported by Research Program Z1-5434 and Research Project BI-US/14-15-026 of the Slovenian Research Agency.

§Partially supported by National Science Foundation grant DMS-1223057.

to reconstruct the historical associations between the phylogenies of host and parasite under a model of parasites switching hosts, which is an instance of the more general problem of *cophylogeny estimation*. See [19, 20] for applications in biology. Diaconis and Holmes have previously demonstrated how one can encode a phylogenetic tree as a series of binary matchings [7], which is a distinct use of matchings from that discussed here.

In computer science, the Tanglegram Layout Problem (TL) is to find a drawing of a tanglegram in the plane with the left and right trees both given as planar embeddings with the smallest number of crossings among (straight) edges matching the leaves of the left tree and the right tree [3]. These authors point out that tanglegrams occur in the analysis of software projects and clustering problems.

In this paper, we give the exact enumeration of tanglegrams with n matched pairs of vertices, along with a simple asymptotic formula and an algorithm for choosing a tanglegram uniformly at random. We refer to the number of matched vertices in a tanglegram as its *size*. Furthermore, two tanglegrams are considered to be equivalent if one is obtained from the other by replacing the tree on the left or the tree on the right by isomorphic trees. For example, in Figure 1, the two non-equivalent tanglegrams of size 3 are shown.

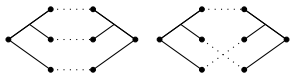

Fig. 1: The tanglegrams of size 3.

We state our main results here postponing some definitions until Section 2. The following is our main theorem.

Theorem 1. *The number of tanglegrams of size n is*

$$t_n = \sum_{\lambda} \frac{\prod_{i=2}^{\ell(\lambda)} \left(2(\lambda_i + \cdots + \lambda_{\ell(\lambda)}) - 1\right)^2}{z_\lambda},$$

where the sum is over binary partitions of n and z_λ is defined by Equation (1)*. Note, if λ has one part, the corresponding empty product in the numerator is 1.*

The first 10 terms of the sequence t_n starting at $n = 1$ are

$$1, 1, 2, 13, 114, 1509, 25595, 535753, 13305590, 382728552,$$

see [18, A258620] for more terms.

Example. The binary partitions of $n = 4$ are (4), $(2, 2)$, $(2, 1, 1)$ and $(1, 1, 1, 1)$, so

$$t_4 = \frac{1}{4} + \frac{3^2}{8} + \frac{3^2 \cdot 1^2}{4} + \frac{5^2 \cdot 3^2 \cdot 1^2}{24} = 13$$

as shown in Figure 2. It takes a computer only a moment to compute

$$t_{42} = 338891364203784804928696774151869483052781762630207228322516215200637 57$$

and under a minute to compute all 3160 integer digits of t_{1000} using a recurrence based on Theorem 1, see Section 5.

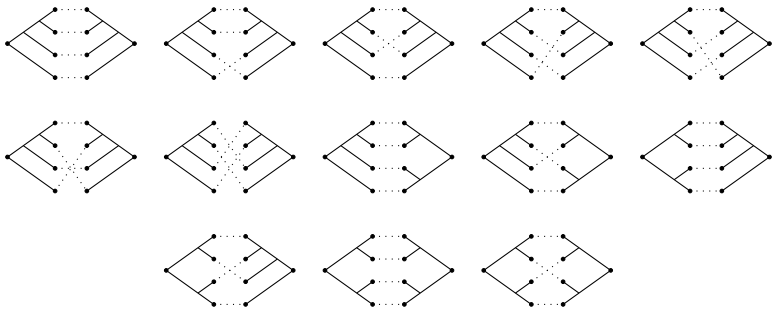

Fig. 2: The 13 tanglegrams of size 4.

We can use the main theorem to study the asymptotics of the sequence t_n.

Corollary 2. *We have*

$$\frac{t_n}{n!} \sim \frac{e^{\frac{1}{8}} 4^{n-1}}{\pi n^3} \qquad \textit{and} \qquad t_n \sim \frac{2^{2n-\frac{3}{2}} \cdot n^{n-\frac{5}{2}}}{\sqrt{\pi} \cdot e^{n-\frac{1}{8}}}.$$

We can also compute approximations of higher degree. For example, we have

$$t_n = \frac{2^{2n-\frac{3}{2}} \cdot n^{n-\frac{5}{2}}}{\sqrt{\pi} \cdot e^{n-\frac{1}{8}}} \cdot \left(1 + \frac{13}{12\,n} + \frac{3089}{2304\,n^2} + \frac{931423}{414720\,n^3} + \frac{826301423}{159252480\,n^4} + \frac{211060350013}{13377208320\,n^5} + O\left(n^{-6}\right)\right).$$

A side result of the proof is a new formula for the number of inequivalent binary trees, called the Wedderburn-Etherington numbers [18, A001190].

Theorem 3. *The number of inequivalent binary trees with n leaves is*

$$b_n = \sum_{\lambda} \frac{\prod_{i=2}^{\ell(\lambda)} (2(\lambda_i + \cdots + \lambda_{\ell(\lambda)}) - 1)}{z_\lambda},$$

where the sum is over binary partitions of n.

A *tangled chain* is an ordered sequence of k binary trees with matchings between neighboring trees in the sequence. For $k = 1$, these are inequivalent binary trees, and for $k = 2$, these are tanglegrams, so the following generalizes Theorems 1 and 3.

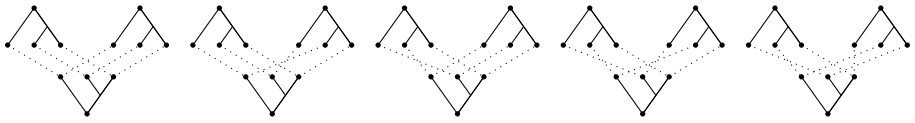

Fig. 3: The tangled chains of length 3 for $n = 3$.

In terms of computational biology, tangled chains of length k formalize the essential input to a variety of problems on k leaf-labeled (phylogenetic) trees (e.g. [22]).

Theorem 4. *The number of ordered tangled chains of length k for n is*

$$\sum_{\lambda} \frac{\prod_{i=2}^{\ell(\lambda)} \left(2(\lambda_i + \cdots + \lambda_{\ell(\lambda)}) - 1\right)^k}{z_\lambda},$$

where the sum is over binary partitions of n.

From the enumerative point of view, it is also quite natural to ask how likely a particular tree T is to appear on one side or the other of a uniformly selected tanglegram. In Section 6, we give a simple explicit conjecture for the asymptotic growth of the expected number of copies of T on one side of a tanglegram as a function of T and the size of the tanglegram. For example, the cherries of a binary tree are pairs of leaves connected by a common parent. We conjecture that the expected number of cherries in one of the binary trees of a tanglegram of size n chosen in the uniform distribution is $n/4$.

Further discussion of the applications of tanglegrams along with several variations on the theme are described in [17]. In particular, tanglegrams can be used to compute the subtree-prune-regraft distance between two binary trees. In a recent follow up paper, Gessel has used the formula given here for binary trees to count several variations on tanglegrams using the theory of species [11].

Gessel also noted that our formula for binary trees can be interpreted as an instance of Burnside's lemma. Let $\mathfrak{S}_n$ act on leaf labeled binary trees with n leaves by permuting the labels. The number of fixed points of $w \in \mathfrak{S}_n$ under this action only depends on the cycle type of w. If we multiply and divide our formula by $n!$, then $n!/z_\lambda$ counts the number of permutations in $\mathfrak{S}_n$ with cycle type λ. Hence, the product corresponding to a binary partition λ counts the number of fixed points of a permutation w with type λ. If w has cycle type which is not a binary partition then w has no fixed trees under this action. Similar reasoning can be applied to $\mathfrak{S}_n$ actions on pairs of trees to relate the formula for tanglegrams to fixed points, and this extends to tangled chains. This proves the following corollary.

Corollary 5. *The product $\prod_{i=2}^{\ell(\lambda)} \left(2(\lambda_i + \cdots + \lambda_{\ell(\lambda)}) - 1\right)^k$ counts the number of fixed points of any permutation $w \in S_n$ of cycle type λ, a binary partition of n, acting on ordered labeled tangled chains of size n and length k.*

The extended abstract proceeds as follows. In Section 2, we define our terminology. We sketch the proof of the main theorems in Section 3. Section 4 contains an algorithm to choose a tanglegram uniformly at random for a given n and we give an asymptotic approximation for the number of tanglegrams. We conclude with several open problems and conjectures in Section 6.

The full version of the paper is [2]. Several papers continuing the study of trees, tanglegrams and tangled chains have recently appeared on the arXiv including [6, 10, 11, 16].

2 Background

In this section, we recall some vocabulary and notation on partitions and trees. This terminology can also be found in standard textbooks on combinatorics such as [21]. We use these terms to give the formal definition of tanglegrams and the notation used in the main theorems.

A *partition* $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_k)$ is a weakly decreasing sequence of positive integers. The length $\ell(\lambda)$ of a partition is the number of entries in the sequence, and $|\lambda|$ denotes the sum of the entries of λ. We say λ is a *binary partition* if all its parts are equal to a nonnegative power of 2. Binary partitions have

appeared in a variety of contexts, see for instance in [14, 15] and [18, A000123]. When writing partitions, we sometimes omit parentheses and commas.

If λ is a nonempty binary partition with m_i occurrences of the letter 2^i for each i, we also denote λ by $(1^{m_0}, 2^{m_1}, 4^{m_2}, 8^{m_3}, \ldots, (2^j)^{m_j})$ where $2^j = \lambda_1$ is the maximum value in λ. Given $\lambda = (1^{m_0}, 2^{m_1}, \ldots, (2^j)^{m_j})$, let z_λ denote the product

$$z_\lambda = 1^{m_0} 2^{m_1} \cdots (2^j)^{m_j} m_0! m_1! m_2! \cdots m_j!. \tag{1}$$

The numbers z_λ are well known since the number of permutations in $\mathfrak{S}_n$ with cycle type λ is $n!/z_\lambda$ [21, Prop. 1.3.2]. For example, for $\lambda = 44211 = (1^2, 2^1, 4^2)$, $z_\lambda = 1^2 \cdot 2^1 \cdot 4^2 \cdot 2! \cdot 1! \cdot 2! = 128$.

A tree is a graph with no cycles; some experts call this a non-plane tree since the embedding in the plane is irrelevant. A rooted tree has one distinguished vertex assumed to be a common ancestor of all other vertices. The neighbors of the root are its *children*. Each vertex other than the root has a unique parent going along the path back to the root, the other neighbors are its children. In a binary tree, each vertex either has two children or no children. A vertex with no children is a *leaf*, and a vertex with two children is an *internal vertex*.

Two binary rooted trees with distinct labeled leaves are said to be *equivalent* if there is an isomorphism from one to the other as graphs mapping the root of one to the root of the other. Let B_n be the set of inequivalent binary rooted trees with $n \geq 1$ leaves, and let b_n be the number of elements in the set B_n. The sequence of b_n's for $n \geq 1$ begins

$$1, 1, 1, 2, 3, 6, 11, 23, 46, 98.$$

We can inductively define a linear order on rooted trees as follows. We say that $T > S$ if either:

- T has more leaves than S
- T and S have the same number of leaves, T has subtrees T_1 and T_2, $T_1 \geq T_2$, S has subtrees S_1 and S_2, $S_1 \geq S_2$, and $T_1 > S_1$ or $T_1 = S_1$ and $T_2 > S_2$

We assume that every tree T in B_n, $n \geq 2$, is presented so that $T_1 \geq T_2$, where T_1 is the left subtree (or upper subtree if the tree is drawn with the root on the left or on the right) and T_2 is the right (or lower) subtree.

Each tree $T \in B_n$ represents a distinct S_n orbit on leaf labeled binary trees with n-leaves. We can define its automorphism group $A(T)$ as follows. Fix a labeling on the leaves of T using the numbers $1, 2, \ldots, n$. Label each internal vertex by the union of the labels for each of its children. The edges in T are pairs of subsets from $[n] := \{1, \ldots, n\}$, each representing the label of a child and its parent. Let $v = [v(1), v(2), \ldots, v(n)]$ be a permutation in the symmetric group $\mathfrak{S}_n$. Then, $v \in A(T)$ if permuting the leaf labels by the function $i \mapsto v(i)$ for each i leaves the set of edges fixed.

A theorem from [13] tells us that if T is a tree with subtrees T_1 and T_2, then $A(T)$ is isomorphic to $A(T_1) \times A(T_2)$ if $T_1 \neq T_2$, and to the wreath product $A(T_1) \wr \mathbb{Z}_2$ if $T_1 = T_2$. Since the automorphism group of a tree on one vertex is trivial, this implies that the general $A(T)$ can be obtained from copies of $\mathbb{Z}_2$ by direct and wreath products (see [17] for more details). Furthermore, if $T_1 \neq T_2$, then the conjugacy type of an element of $A(T)$ is $\lambda^1 \cup \lambda^2$, where λ^i is the conjugacy type of an element of $A(T_i)$, $i = 1, 2$, and $\lambda^1 \cup \lambda^2$ is the multiset union of the two sequences written in decreasing order. If $T_1 = T_2$, then for an arbitrary element of $A(T)$ either the leaves in each subtree remain in that subtree, or all leaves are

mapped to the other subtree. The conjugacy type of an element of $A(T)$ is then either $\lambda^1 \cup \lambda^2$, where λ^i is the conjugacy type of an element of $A(T_i)$, $i = 1, 2$, or it is $2\lambda^1$, where λ^1 is the conjugacy type of an element of $A(T_1)$. In particular, the conjugacy type of any element of the automorphism group of a binary tree must be a binary partition.

Next, we define tanglegrams. Given a permutation $v \in \mathfrak{S}_n$ along with two trees $T, S \in B_n$ each with leaves labeled $1, \ldots, n$, we construct an *ordered binary rooted tanglegram* (T, v, S) of size n with T as the left tree, S as the right tree, by identifying leaf i in T with leaf $v(i)$ in S. Note, (T, v, S) and (T', v', S') are considered to represent the same tanglegram provided $T = T'$, $S = S'$ as trees and $v' = uvw$ where $u \in A(T)$ and $w \in A(S)$. Let T_n be the set of all ordered binary rooted tanglegrams of size n, and let t_n be the number of elements in the set T_n. For example, $t_3 = 2$ and $t_4 = 13$. Figures 1 and 2 show the tanglegrams of sizes 3 and 4 where we draw the leaves of the left and right tree on separate vertical lines and show the matching using dotted lines. The dotted lines are not technically part of the graph, but this visualization allows us to give a planar drawing of the two trees.

We remark that the *plane binary trees* with $n \geq 2$ leaves are a different family of objects from B_n that also come up in this paper. These are trees embedded in the plane so the left child of a vertex is distinguishable from the right child. The plane binary trees with $n + 1$ leaves are well known to be counted by Catalan numbers

$$c_n = \frac{1}{n+1}\binom{2n}{n} = \frac{2^n(2n-1)!!}{(n+1)!}$$

because they clearly satisfy the Catalan recurrence $c_n = c_0c_{n-1} + c_1c_{n-2} + c_2c_{n-3} + \cdots + c_{n-1}c_0$ with $c_0 = c_1 = 1$. For example, there are $c_2 = 2$ distinct plane binary trees with 3 leaves which are mirror images of each other while $b_3 = 1$.

Dulucq and Guibert [8] have studied "twin binary trees", which are pairs of plane binary trees with certain matched vertices. This is the plane version of tanglegrams. They show that twin binary trees are in bijection with Baxter permutations. The Baxter permutations in $\mathfrak{S}_n$ are enumerated by a formula due to Chung-Graham-Hoggart-Kleiman [5]

$$a_n = \frac{\sum_{k=1}^{n} \binom{n+1}{k-1}\binom{n+1}{k}\binom{n+1}{k+1}}{\binom{n+1}{1}\binom{n+2}{2}}.$$

3 Sketch of proof of the main theorem

The focus of this section is the proof of Theorem 1. The theorem will follow from a auxiliary result, and the proof of Theorem 4 is similar and is omitted in this extended abstract.

The number of tanglegrams is, by definition, equal to

$$t_n = \sum_T \sum_S |\mathcal{C}(T, S)|,$$

where the sums on the right are over inequivalent binary trees with n leaves, and $\mathcal{C}(T, S)$ is the set of double cosets of the symmetric group $\mathfrak{S}_n$ with respect to the double action of $A(T)$ on the left and $A(S)$ on the right. See [17] for more details. Let us fix $T \in B_n$ and $S \in B_n$ and write $\mathcal{C} = \mathcal{C}(T, S)$. Then

$$|\mathcal{C}| = \sum_{C \in \mathcal{C}} 1 = \sum_{C \in \mathcal{C}} \frac{|C|}{|C|} = \sum_{C \in \mathcal{C}} \sum_{w \in C} \frac{1}{|C|} = \sum_{w \in \mathfrak{S}_n} \frac{1}{|C_w|},$$

where C_w is the double coset of $\mathfrak{S}_n$ that contains w. It is known (e.g. [12, Theorem 2.5.1 on page 45 and Exercise 40 on page 49]) that the size of the double coset $C_w = A(T)wA(S)$ is the quotient

$$\frac{|A(T)| \cdot |A(S)|}{|A(T) \cap wA(S)w^{-1}|},$$

and therefore,

$$|\mathcal{C}| = \sum_{w \in \mathfrak{S}_n} \frac{|A(T) \cap wA(S)w^{-1}|}{|A(T)| \cdot |A(S)|}.$$

We have

$$\sum_{w \in \mathfrak{S}_n} |A(T) \cap wA(S)w^{-1}| = \sum_{w \in \mathfrak{S}_n} \sum_{u \in A(T)} \sum_{v \in A(S)} [u = wvw^{-1}] = \sum_{u \in A(T)} \sum_{v \in A(S)} \sum_{w \in \mathfrak{S}_n} [u = wvw^{-1}],$$

where $[\cdot]$ is the indicator function. Now $u = wvw^{-1}$ can only be true if u and v are permutations of the same conjugacy type λ, which must necessarily be a binary partition as noted above. Furthermore, if u and v are both of type λ, then there are z_λ permutations w for which $u = wvw^{-1}$. That means that

$$|\mathcal{C}(T,S)| = \frac{\sum_\lambda |A(T)_\lambda| \cdot |A(S)_\lambda| \cdot z_\lambda}{|A(T)| \cdot |A(S)|}, \tag{2}$$

where $A(T)_\lambda$ (respectively, $A(S)_\lambda$) denotes the elements of $A(T)$ (resp., $A(S)$) of type λ.

To get the formula for t_n we want to sum Equation (2) over all pairs of trees, and fortunately a change of the order of summation helps. Indeed, we have

$$t_n = \sum_T \sum_S \frac{\sum_\lambda |A(T)_\lambda| \cdot |A(S)_\lambda| \cdot z_\lambda}{|A(T)| \cdot |A(S)|} = \sum_\lambda z_\lambda \cdot \sum_T \sum_S \frac{|A(T)_\lambda| \cdot |A(S)_\lambda|}{|A(T)| \cdot |A(S)|} \tag{3}$$

$$= \sum_\lambda z_\lambda \cdot \left(\sum_T \frac{|A(T)_\lambda|}{|A(T)|} \right)^2, \tag{4}$$

and the main theorem is proved once we have shown the following proposition.

Proposition 6. *For a binary partition λ,*

$$\sum_{T \in B_n} \frac{|A(T)_\lambda|}{|A(T)|} = \frac{\prod_{i=2}^{\ell(\lambda)} (2\left(\lambda_i + \cdots + \lambda_{\ell(\lambda)}\right) - 1)}{z_\lambda},$$

where $A(T)_\lambda$ denotes the elements of $A(T)$ of type λ.

The proposition also implies Theorem 3, as

$$\sum_T 1 = \sum_T \sum_\lambda \frac{|A(T)_\lambda|}{|A(T)|} = \sum_\lambda \sum_T \frac{|A(T)_\lambda|}{|A(T)|}.$$

If $\lambda = 1^n$, then $|A(T)_\lambda| = 1$ for all $T \in B_n$, so the proposition is saying that

$$\sum_T \frac{1}{|A(T)|} = \frac{(2n-3)!!}{n!} = \frac{c_{n-1}}{2^{n-1}}.$$

This is equivalent to $\sum_T 2^{n-1}/|A(T)| = c_{n-1}$. Since $2^{n-1}/|A(T)|$ counts all plane binary trees isomorphic to T, this is just the well-known fact that there are c_{n-1} plane binary trees with n leaves.

For a general λ, however, the proposition is far from obvious. What we need is a recursion satisfied by the expression on the right, analogous to the recursion $c_n = c_0c_{n-1} + c_1c_{n-1} + \cdots + c_{n-1}c_0$ for Catalan numbers.

Lemma 7. *For a nonempty subset $S = \{i_1 < i_2 < \ldots < i_k\}$ of the natural numbers define*

$$r_S(x_1, x_2, \ldots) = (x_{i_2} + \cdots + x_{i_k} - 1)(x_{i_3} + \cdots + x_{i_k} - 1)\cdots(x_{i_{k-1}} + x_{i_k} - 1)(x_{i_k} - 1). \qquad (5)$$

Let $n \geq 2$, let $\mathbf{x}$ denote variables $x_1, x_2, \ldots$, and let $\mathbf{x}/2$ denote $x_1/2, x_2/2, \ldots$. Then

$$r_{[n]}(\mathbf{x}) = 2^{n-1} r_{[n]}(\mathbf{x}/2) + \sum_{1 \in S \subsetneq [n]} r_S(\mathbf{x}) \cdot r_{[n]\setminus S}(\mathbf{x}).$$

The proof is by induction on n. See [2] for complete details.

Example. For $n = 3$, the lemma says that

$$(x_2 + x_3 - 1)(x_3 - 1) = (x_2 + x_3 - 2)(x_3 - 2) + 1 \cdot (x_3 - 1) + (x_2 - 1) \cdot 1 + (x_3 - 1) \cdot 1,$$

where the last three terms on the right-hand side correspond to subsets $\{1\}$, $\{1, 2\}$, and $\{1, 3\}$, respectively. As another example, take $x_i = 2$ for all i. Then $r_S(\mathbf{x}) = (2|S| - 3)!!$ (where we interpret $(-1)!!$ as 1), $r_S(\mathbf{x}/2) = 0$, and by the obvious symmetry of S and $[n] \setminus S$ the lemma yields

$$2 \cdot (2n - 3)!! = \sum_{k=1}^{n-1} \binom{n}{k} (2k - 3)!!(2n - 2k - 3)!!,$$

which is equivalent to the standard recurrence for Catalan numbers.

Proof of Proposition 6. Say λ is a binary partition of n. The proof is by induction on n. For $n = 1$, the statement is obvious. Assume that the statement holds for all binary partitions up to size $n - 1$. Our task is to show

$$\sum_T \frac{|A(T)_\lambda|}{|A(T)|} = \frac{r_{[\ell(\lambda)]}(2\lambda_1, 2\lambda_2, 2\lambda_3, \ldots)}{z_\lambda}$$

by showing the left hand side satisfies a recurrence similar to (5). This can be done by a careful analysis of all possible cases and is omitted in this extended abstract. □

4 Random generation of tanglegrams

Algorithm 1 (Random generation of $w \in A(T)$)**.**

Input: Binary tree $T \in B_n$.

Procedure: If T is the tree with one vertex, let w be the unique element of $A(T)$. Otherwise, the root of T has subtrees T_1 and T_2. Assume the leaves of T_1 are labeled $[1, k]$ and the leaves of T_2 are labeled $[k + 1, n]$. Use the algorithm recursively to produce $w_i \in A(T_i)$, $i = 1, 2$ where $A(T_1)$ is a subset of the permutations of $[1, n]$ which fix $[k + 1, n]$ and $A(T_2)$ is a subset of the permutations of $[1, n]$ which fix $[1, k]$. Construct w as follows. Say $f : [1, k] \longrightarrow [k + 1, n]$ mapping i to $i + k$ induces an isomorphism of T_1 and T_2. Define the "tree flip permutation" π to be the product of the transpositions interchanging i with $f(i)$ for all $1 \leq i \leq k$.

- If $T_1 \neq T_2$, set $w = w_1 w_2$.
- If $T_1 = T_2$, choose either $w = w_1 w_2$ or $w = \pi w_1 w_2$ with equal probability.

Output: Permutation $w \in A(T)$.

Algorithm 2 (Random generation of T with non-empty $A(T)_\lambda$ and $w \in A(T)_\lambda$)**.**

Input: Binary partition λ of n.

Procedure: If $n = 1$, let T be the tree with one vertex, and let w be the unique element of $A(T)$. Otherwise, pick a subdivision (λ^1, λ^2) from $\{(\lambda^1, \lambda^2)\colon \lambda^1 \cup \lambda^2 = \lambda\} \cup \{(\lambda/2, \lambda/2)\}$, where (λ^1, λ^2) is chosen with probability proportional to $q_{\lambda^1} q_{\lambda^2}$ and $(\lambda/2, \lambda/2)$ with probability proportional to $q_{\lambda/2}$, where $t_n = \sum z_\lambda q_\lambda^2$.

- If $\lambda^1, \lambda^2 \neq \lambda/2$, use the algorithm recursively to produce trees T_1, T_2 and permutations $w_1 \in A(T_1)_{\lambda^1}$, $w_2 \in A(T_2)_{\lambda^2}$. If necessary, switch $T_1 \leftrightarrow T_2$, $w_1 \leftrightarrow w_2$ so that $T_1 \geq T_2$. Let $T = (T_1, T_2)$, $w = w_1 w_2$.
- If $\lambda^1 = \lambda^2 = \lambda/2$, use the algorithm recursively to produce a tree T_1 and a permutation $w_2 \in A(T_1)_{\lambda/2}$, and use Algorithm 1 to produce a permutation $w_1 \in A(T_1)$. Let $T = (T_1, T_1)$ and $w = \pi w_1 \pi w_1^{-1} \pi w_2$.

Output: Binary tree T and permutation $w \in A(T)_\lambda$.

Algorithm 3 (Random generation of tanglegrams)**.**

Input: Integer n.
Procedure: Pick a random binary partition λ of n with probability proportional to $z_\lambda q_\lambda^2$ where $t_n = \sum z_\lambda q_\lambda^2$. Use Algorithm 2 twice to produce random trees T and S and permutations $u \in A(T)_\lambda$, $v \in A(S)_\lambda$. Among the permutations w for which $u = wvw^{-1}$, pick one at random from the z_λ possibilities.

Output: Binary trees T and S and double coset $A(T)wA(S)$, or equivalently (T, w, S).

Algorithm 4 (Random generation of $T \in B_n$)**.** Algorithm 4 is not the first of its kind, see also [9].

Input: Integer n.
Procedure: Pick a random binary partition λ of n with probability proportional to q_λ. Use Algorithm 2 to produce a random tree T (and a permutation $u \in A(T)_\lambda$).

Output: Binary tree T.

Algorithm 5 (Random generation of tangled chains)**.**

Input: Positive integers k and n.

Procedure: Pick a random binary partition λ of n with probability proportional to $z_\lambda^{k-1} q_\lambda^k$ where $t(k, n) = \sum z_\lambda^{k-1} q_\lambda^k$. Use Algorithm 2 k times to produce random trees T_i and permutations $u_i \in A(T_i)_\lambda$ for $i = 1, \ldots, k$. Among the permutations w_i for which $u_i = w_i u_{i+1} w_i^{-1}$, pick one uniformly at random for each $i = 1, \ldots, k-1$.

Output: $(T_1, \ldots, T_k)$ and $(w_1, \ldots, w_{k-1})$.

Theorem 8. *For any positive integer n, the following hold. Algorithm 1 produces every permutation $w \in A(T)$ with probability $\frac{1}{|A(T)|}$. Algorithm 2 produces every pair (T, w), where $w \in A(T)_\lambda$, with probability $\frac{1}{|A(T)|\cdot q_\lambda}$. Algorithm 3 produces every tanglegram with probability $\frac{1}{t_n}$. Algorithm 4 produces every inequivalent binary tree with probability $\frac{1}{b_n}$. Algorithm 5 produces every tangled chain of length k of trees in B_n with probability $\frac{1}{t(k,n)}$.*

5 A recurrence for enumerating tanglegrams and tangled chains

In this section, we give a recurrence for computing t_n. Recall that for each nonempty binary partition λ, we can construct its *multiplicity vector* $m^\lambda = (m_0, m_1, m_2, m_3, \ldots)$ where m_i is the number of times 2^i occurs in λ. The map $\lambda \mapsto m^\lambda$ is a bijection from binary partitions to vectors of nonnegative integers with only finitely many nonzero entries. The quantity z_λ for a binary partition λ is easily expressed in terms of the multiplicities in m^λ as

$$z_\lambda = \prod_{h\geq 0} 2^{h\cdot m_h}\ m_h! = \prod_{\substack{h\geq 0\\ m_h\neq 0}} \prod_{j=1}^{m_h} j\cdot 2^h$$

We will use the functions $f^2(s) := (2s-1)^2,\ c(h,m,s) := \prod_{j=1}^{m} \frac{f^2(s+j\cdot 2^h)}{j\cdot 2^h}$, and

$$r(h,n,s) := \sum_{\substack{m=0\\ (n-m)\text{ even}}}^{n} c(h,m,s)\, r\left(h+1, \frac{n-m}{2}, s+m2^h\right) \tag{6}$$

with base cases $c(h,0,s) = r(h,0,s) = 1$.

Theorem 9. *For $n \geq 1$, the number of tanglegrams is $t_n = \frac{r(0,n,0)}{f^2(n)}$, which can be computed recursively using* (6).

The general case is spelled out in [2]. The proof is a direct consequence of Theorem 1. Similar recurrence relations hold for all tangled chains.

6 Final remarks

Variants on tanglegrams

Tanglegrams as described here fit in a set of more general setting of pairs of graphs with a bijection between certain subsets of the vertices (more completely described and motivated in [17]). One can also consider *unordered tanglegrams* by identifying (T, v, S) with (S, v^{-1}, T). For example, the 4th and 5th tanglegrams in Figure 2 are equivalent as unordered tanglegrams, and so are the 8th and 10th. From this picture, the reader can verify that there are 10 unordered tanglegrams of size 4.

Because of reversibility assumptions for the continuous time Markov mutation models commonly used to reconstruct phylogenetic trees, unrooted trees are the most common output of phylogenetic inference algorithms. Thus another variant of tanglegrams involves using unrooted trees in place of rooted ones. The motivation for studying these variants comes from noting that many problems in computational phylogenetics such as distance calculation between trees [1] "factor" through a problem on tanglegrams.

Connection with symmetric functions

The main theorems suggest that symmetric functions might be at play; note, for example, the similarity with the formula $h_n = \sum_\lambda z_\lambda^{-1} p_\lambda$, where h_n is the homogeneous symmetric function, p_λ the power sum symmetric function, and the sum is over all partitions of n. Is there a connection between tanglegrams (or more generally tangled chains) and symmetric functions?

Based on a manuscript version of this paper, Ira Gessel pointed out that there is indeed a connection between symmetric functions and the enumeration of tanglegrams based on the theory of species. He has beautifully spelled out this connection. This approach leads to a simple formula for the number of unordered tanglegrams and a generating function for the number of unrooted tanglegrams along with several other variations on tanglegrams [11].

Alternative proofs

Recently, Eric Fusy gave a combinatorial proof of our main results, which also yields a remarkable simplification of the random sampler for tangled chains [10].

The shape of a random tanglegram

Given an algorithm for random generation, it is natural to ask for the probability of certain substructures in trees, tanglegrams and tangled chains. For example, cherries (two leaves with a common parent) play an important role in the literature on tanglegrams, see [4, pp. 325–326]. In the original version of this abstract and the corresponding full length paper, we stated several open problems and conjectures on the limiting distribution of certain substructures. Many of these problems have now been solved by Konvalinka and Wagner [16] and Czabarka, Székely, and Wagner [6]. In particular, Konvalinka and Wagner show that the two halves of a random tanglegram essentially look like two independently chosen random plane binary trees.

Acknowledgements

We thank Ira Gessel, Arnold Kas, Jim Pitman, Xavier G. Viennot, Paul Viola, Bianca Viray, Stephan Wagner, and Chris Whidden for helpful discussions. We also thank two anonymous referees for their insightful suggestions.

References

[1] B. L. ALLEN AND M. STEEL, *Subtree transfer operations and their induced metrics on evolutionary trees*, Ann. Comb., 5 (2001), pp. 1–15.

[2] S. C. BILLEY, M. KONVALINKA, AND F. A. MATSEN IV, *On the enumeration of tanglegrams and tangled chains*, arXiv:1507.04976, (2015).

[3] K. BUCHIN, M. BUCHIN, J. BYRKA, M. NÖLLENBURG, Y. OKAMOTO, R. I. SILVEIRA, AND A. WOLFF, *Drawing (complete) binary tanglegrams: hardness, approximation, fixed-parameter tractability*, Algorithmica, 62 (2012), pp. 309–332.

[4] M. A. CHARLESTON, *Recent results in cophylogeny mapping*, in The Evolution of Parasitism-A phylogenetic perspective, T. Littlewood, ed., vol. 54 of Advances in Parasitology, Academic Press, 2003, pp. 303 – 330.

[5] F. R. K. CHUNG, R. L. GRAHAM, V. E. HOGGATT, JR., AND M. KLEIMAN, *The number of Baxter permutations*, J. Combin. Theory Ser. A, 24 (1978), pp. 382–394.

[6] É. CZABARKA, L. A. SZÉKELY, AND S. WAGNER, *Inducibility in binary trees and crossings in random tanglegrams*, ArXiv e-prints, (2016).

[7] P. W. DIACONIS AND S. P. HOLMES, *Matchings and phylogenetic trees*, Proc. Natl. Acad. Sci. U. S. A., 95 (1998), pp. 14600–14602.

[8] S. DULUCQ AND O. GUIBERT, *Permutations de Baxter*, Sém. Lothar. Combin., 33 (1994), pp. Art. B33c, approx. 8 pp.˙ 33. Tagung des Lotharingischen Kombinatorikseminars (Freiberg, 1994).

[9] G. W. FURNAS, *The generation of random, binary unordered trees*, J. Classification, 1 (1984), pp. 187–233.

[10] É. FUSY, *On symmetries in phylogenetic trees*, ArXiv e-prints, (2016).

[11] I. M. GESSEL, *Counting tanglegrams with species*, ArXiv e-prints, (2015).

[12] I. N. HERSTEIN, *Topics in algebra*, Xerox College Publishing, Lexington, Mass.-Toronto, Ont., second ed., 1975.

[13] C. JORDAN, *Sur les assemblages de lignes*, J. Reine Angew. Math., (1869), pp. 185–190.

[14] D. E. KNUTH, *Correction: "An almost linear recurrence"*, Fibonacci Quart, 4 (1966), p. 354.

[15] M. KONVALINKA AND I. PAK, *Cayley compositions, partitions, polytopes, and geometric bijections*, J. Combin. Theory Ser. A, 123 (2014), pp. 86–91.

[16] M. KONVALINKA AND S. WAGNER, *The shape of random tanglegrams*, ArXiv e-prints, (2015).

[17] F. A. MATSEN IV, S. C. BILLEY, A. KAS, AND M. KONVALINKA, *Tanglegrams: a reduction tool for mathematical phylogenetics*, arXiv preprint, (2015).

[18] OEIS FOUNDATION INC., *The On-Line Encyclopedia of Integer Sequences*, 2015. `http://oeis.org`.

[19] R. D. PAGE, *Tangled trees : phylogeny, cospeciation, and coevolution / ed. by Roderic D.M. Page*, Chicago [etc.] : The University of Chicago Press, 2003.

[20] C. SCORNAVACCA, F. ZICKMANN, AND D. H. HUSON, *Tanglegrams for rooted phylogenetic trees and networks*, Bioinformatics, 27 (2011), pp. i248–i256.

[21] R. P. STANLEY, *Enumerative Combinatorics. Vol. 1*, vol. 49 of Cambridge Studies in Advanced Mathematics, Cambridge University Press, Cambridge, 1997.

[22] C. WHIDDEN, N. ZEH, AND R. G. BEIKO, *Supertrees based on the subtree prune-and-regraft distance*, Syst. Biol., (2014).