Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

On The Exact Counting of Tree-Child Networks

Miquel Pons (m.pons-viver@uib.es)
University of the Balearic Islands
Josep Batle
University of the Balearic Islands

Research Article

Keywords: Evolutionary histories, phylogenetic networks, combinatorial study

Posted Date: July 1st, 2021

DOI: https://doi.org/10.21203/rs.3.rs-605996/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

On the exact counting of tree-child networks

Miquel Pons[1,*] Josep Batle[1]

[1]Universitat de les Illes Balears. Departament de Física.
Palma de Mallorca 07122. Balearic Islands, Spain.
[*]Corresponding author: m.pons-viver@uib.es

June 28, 2021

Abstract

The combinatorial study of phylogenetic networks has attracted much attention in recent times. In particular, one class of them, the so-called *tree-child networks*, are becoming the most prominent ones. However, their combinatorial properties are largely unknown. In this paper we address the problem of exactly counting them. We conjecture a bijection with a certain class of words, and from this assumption a simple recurrence formula arises. It is able to determine the number of all subclasses, as well as a direct formula, a simple enumeration procedure and precise asympotics. Our results coincide with all currently proved formulas for particular subclasses of *tree-child networks*, as well as with numerical results obtained for small networks. Since, as we will show, working with words greatly simplifies the problem, we expect to contribute to further combinatoric characterizations of this class of networks.

Introduction

Evolutionary histories of several kinds are usually represented with the mathematical help of phylogenetic trees. Linguistics, but mostly genomics are the traditional areas where this tool is employed. However, mechanisms of reticulate evolution, such as horizontal gene hybridization, transfer or recombination, render such trees less appropriate. When the species involved in those events have more than one ancestor, *phylogenetic networks* are better suited [18, 16, 21, 17]. Also, the comparison of phylogenetic trees and networks is attackting considerable attention [2, 6]. Quite recently, a phylogenetic network of SARS-CoV-2 genomes was sampled from across the world in order to better understand the outbreak of the ongoing Covid-19 coronavirus world pandemic [9].

Due to the increasing popularity of the usage of phylogenetic networks, a combinatorial approach to their study regarding counting, enumeration and stochastic characterization has brought much attention recently [5, 19, 10, 11, 12, 13, 23, 3, 15, 1]. It is usual to impose further restrictions to the general structure of phylogenetic networks (in general they are *labeled directed acyclic graphs*) in order to make them more manageable. Among all classes of phylogenetic networks, we shall study the tree-child networks (TCNs) [4], in which every non-leaf node has a child that is a tree node or a leaf. This class, besides of being biologically justified, it is considered to have good mathematical properties, as well as mathematically interesting, even some authors [1] consider them to be "*mathematically intractable*". The present work tries to shed light on the long-sought problem of counting and enumerating this class of networks.

There exists in the literature exact counting results for TCNs with low number, k, of reticulation events, and arbitrary number, n, of leaves. Specifically for $k = 1, 2$ and 3, see Refs. [23, 5, 11] respectively, and for TCNs with the maximum number of reticulation nodes [13]. Our conjectured formula, Eq. (9), coincides with all these results, and it also reproduces particular values obtained by demanding computational procedures [23, 5].

Specifically, we shall continue the work of M. Fuchs et al. [13] who made use of a similarity from the number of maximally reticulated tree-child networks (we will show it occurs whenever $k = n - 1$) with a certain class of words defined in the *On-Line Encyclopedia of Integer Sequences*, specifically the sequence OEIS A213863. They are simply related by a $n!$ factor. We have been able to obtain the cardinalities of all subclasses $\mathcal{TC}_{n,k}$ of tree-child networks, with arbitrary n and k, by generalizing those words.

In fact, as commented by Flajolet and Sedgewick [8, p.62], words can, at least in principle, encode any combinatorial structure. A classical example is the encoding of set partitions. How to encode them can be found in the same reference, where the translation makes possible an easy counting of the set partitions. A kind of set partition called *binary total partition*, which consists in repeatedly dividing the blocks of an original set into exactly two blocks, until only singletons remain, is bijectively related with the class of binary labeled trees, that is, *phylogenetic trees* (see example 5.2.6 in Ref. [20]).

The outline of the this work goes as follows. We start with a preliminary section, providing general and basic terminology and elementary properties, firstly of general phylogenetic networks and secondly concerning the subclass of tree-child networks. In section III we exhibit the parallelism between TCNs and words. We start by reproducing the bijection given by Fuchs et al. [13] for the maximally reticulated subclass. In the second part of the section we establish a bijection between phylogenetic trees and a class of very similar words. To the best knowledge of our recollection, the encoding induced by the bijection is a brand new one. We argue its practical aspects and also provide a simple algorithm which is a useful tool to generate the entire sequence of words with a minimum difference between any two consecutive ones, useful for exhaustive combinatorial studies of phylogenetic trees. Finally the generalized words are defined, and the relationship between them and arbitrary subclasses of TCNs is conjectured. In section IV, based on the conjecture, counting formulas for TCNs are derived, enumeration procedures are provided and an asymptotic formula is obtained. Hints for the bijection are also presented.

1 Preliminaries

In this section the basic terminology is introduced, including the formal definitions of *phylogenetic networks* and the *tree-child networks subclass*. Elementary, although important properties of these structures are also provided.

1.1 Phylogenetic networks

A *phylogenetic network* $\mathcal{N}$ on X is a rooted acyclic digraph with no edges in parallel satisfying the following properties:

(i) the root has in-degree zero and out-degree one.

(ii) a vertex with out-degree zero has in-degree one and it is called a *leaf*. The set of leaves are bijectively labeled with the elements of X.

(iii) all other vertices either have in-degree one and out-degree two, or in-degree two and out degree-one.

The vertices with in-degree two and out-degree one are called *reticulations*, and the vertices with in-degree one and out-degree two are called *tree vertices*. The edges directed into a reticulation are *reticulation edges*, and all other edges are *tree edges*. In particular, a phylogenetic X-tree is a phylogenetic network on X with no reticulations.

Lemma 1. *For any phylogenetic network with n leaves, k reticulation nodes and t tree nodes the following relation holds:*

$$n + k = t + 1\,. \tag{1}$$

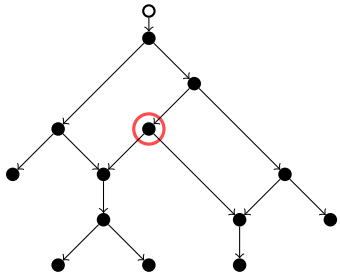
Proof. The sum of the out-degrees is equal to the sum of the in-degrees. □

If u is a vertex of a phylogenetic network $\mathcal{N}$ and (u, v) is an edge in $\mathcal{N}$, we say v is a *child* of u, conversely, u is a *parent* of v. More generally, u is an *ancestor* of a vertex w if there is a directed path from u to w in $\mathcal{N}$, in which case, w is a *descendant* of u.

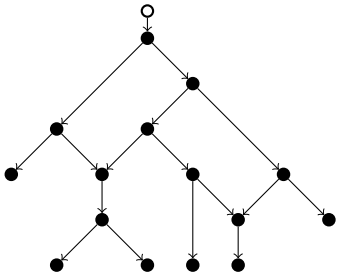
Let $\mathcal{R}(\mathcal{N})$ denote the set of reticulation nodes of the phylogenetic network $\mathcal{N}$. Then, let $\mathcal{N} - \mathcal{R}(\mathcal{N})$ be the subnetwork that is obtained from $\mathcal{N}$ by removing all reticulations together with the incident edges. This subnetwork is actually a forest in which each connected component consists only of tree nodes and it is rooted at either the network root or the child of a reticulation. Each of these connected components is a *tree-component* of $\mathcal{N}$. This is a useful concept for characterizing the topological structures of phylogenetic networks [22, 14].

1.2 Tree-child networks

A phylogenetic network $\mathcal{N}$ on X is a *tree-child network* if each non-leaf vertex v of $\mathcal{N}$ has a child that is either a tree vertex or a leaf. Introduced in Ref. [4], the class of tree-child networks is an increasingly prominent class of phylogenetic networks in the literature. See Figure 1 for some examples. From the definition it follows that no reticulation of $\mathcal{N}$ has a child reticulation and no tree vertex of $\mathcal{N}$ has two child reticulations.



(a) This network is not a TCN because the two childs of the red node (circle) are reticulation nodes.



(b) This network is tree-child.

Figure 1: Examples of Phylogenetic Networks.

It is also said that a tree node is *free* if each of its children is either a tree node or a leaf. Moreover, an edge to a child of a tree node is known as a *free edge*. In the present work we will denote the class of all tree-child networks with n leaves by $\mathcal{TC}_n$, their subclasses having n leaves and k reticulation nodes by $\mathcal{TC}_{n,k}$.

Lemma 2. *[13] Every tree-child network in $\mathcal{TC}_{n,k}$ has $n+k-1$ free tree nodes and thus $2(n-k-1)$ free edges.*

Proof. From (1), we have that a tree-child network from $\mathcal{TC}_{n,k}$ has $n+k-1$ tree nodes. The two parents of every reticulation node are not free, and due to the tree-child property, different reticulation nodes have different parents. Thus, the number of tree nodes which are not free is $2k$, from which the result follows. □

Corollary 2.1. *The number of edges ending either in a tree node or a leaf, called* tree edges*, is equal to $2n+k-1$.*

Proof. Add up all contributions: one edge from the root, k edges from the reticulation nodes, the $2(n-k-1)$ free edges and $2k$ edges from the $2k$ not being free nodes. □

2 Tree-child networks and words

In this section we present bijections between two particular cases of tree-child networks and words. The first considered case is the maximally reticulated TCN. The mapping, to words having every letter repeated exactly three times, was given by Fuchs et al. [13]. The second case is the network without reticulations, that is, a tree. We provide a bijection between trees and a similar class of words, but now every letter is repeated two times. In both situations the word is determined from the network by drawing and labelling non overlapping paths to leaves. We will see that for the maximally reticulated TCN paths do not depend on the label of the leaves, whereas for trees, words depend strongly on leaves' labels. Besides, for trees the mapping is a one-to-one relationship, whereas for the maximally reticulated TCN, due to its asymmetry, $n!$ networks are mapped to the same word, being n the number of leaves. We finalize the section by presenting the driving conjecture: a general TCN, say with k reticulation nodes and n leaves, is a mixture of the above cases, where the associated words have k letters repeated three times and the rest characters are repeated twice.

2.1 Maximally reticulated tree-child networks

We start by introducing the first proposition which gives the fundamental characterization of this particular subclass of TCNs. The property is of paramount importance in order to establish a bijection with words.

Proposition 1. *[13] A tree-child network from $\mathcal{TC}_n$ has $n-1$ reticulation nodes if and only if the path from every node to a leaf whose intermediate nodes are all tree nodes is unique.*

Proof. First, let us assume that we have a tree-child network with n leaves and $n-1$ reticulation nodes. Then, for different reticulation nodes, the paths from these nodes to leaves with all intermediate nodes being tree nodes end with different leaves. Moreover, the child of the root (which is a tree node) also has a path with all intermediate nodes being tree nodes that end with another leaf. Thus, we have already at least n leaves and consequently, no node can have two paths with the claimed property because the number of leaves would thus exceed n.

Next, assume that for every node there is a unique path to a leaf with all intermediate nodes being tree nodes. Consider first this path from the child of the root. Clearly, all intermediate nodes must be parents of reticulation nodes for otherwise an intermediate node would have two different paths to leaves with all intermediate nodes being tree nodes. Moreover, any reticulation node which is the child of an intermediate node on the path is followed by a tree node, which again has a path to a leaf (all intermediate nodes being parents of reticulation nodes). Clearly, this gives a network with n leaves and exactly $n-1$ reticulation nodes. □

Next the class of words is formally defined.

Definition 1. *Let $\mathcal{A}_n$ denote the class of words built from a n-ary alphabet so that in each word w every letter is repeated exactly 3 times, and for every prefix z of w we have $\#(z,a_i)=0$ or $\#(z,a_i)\geq\#(z,a_j)$ for all $j>i$ and the function $\#(z,a_i)$ counts the occurrences of the i-th letter in z.*

The sequence $\{x_n\}_{n\geq 0}=\{1,1,7,106,2575,\ldots\}$ corresponds to the OEIS A213863 entry. Bellow the first classes are shown:

$$\begin{aligned}\mathcal{A}_0 &= \emptyset\\ \mathcal{A}_1 &= \{aaa\}\\ \mathcal{A}_2 &= \{aabbab, ababab, baabab, aaabbb, aababb, abaabb, baaabb\}\end{aligned}$$

The following result was recently discovered by Fuchs et al. [13].

Proposition 2. *There is a bijection from the set of tree-child networks $\mathcal{TC}_{n,n-1}$ with labels removed to $\mathcal{A}_{n-1}$. Consequently, $x_{n-1}=|\mathcal{TC}_{n,k}|/n!$.*

Proof. The bijection goes as follows. Start with whatever network from $\mathcal{TC}_{n,n-1}$ with its labels removed. Recall that due to the lack of symmetry of these networks there are $\mathcal{TC}_{n,n-1}/n!$ in total.

In the first step, we sort the path-components of the chosen tree-child network. We do this inductively. First, the path-component of the child of the root receives an index 0. Assume that k path-components have been indexed. Now, consider all un-indexed path-components whose first node (which is a reticulation node) has its two parents already whithin indexed path components. If both parents are in the same path component, then one is the descendent of the other; call that one the *second parent*; if both parents are in different path components, then the parent in the path-component with the higher index is the *second parent*. Now, sort all the above chosen un-indexed path-components according to the indices of the path-components where the second parents are located and in case their indices coincide, one goes to the ancestor relationship within the path-component of their second parents. Continue this until all path-components are indexed, which will eventually happen because our networks are assumed to be connected.

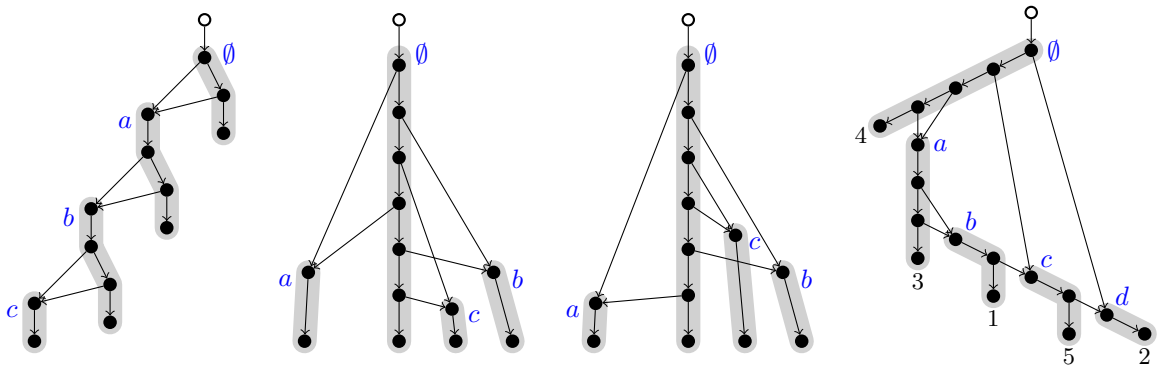
Now, we label the first node of every path-component of index $k>0$ together with its two parents by a_k.

Finally, we read the labels of each path-component starting with the 0-th one until reaching the last one; see Figure 2, where a line separates the strings from different path components, although they are not necessary (they never occur just before the third appearance of a letter).

The resulting word uses $n-1$ letters, $a_1,\ldots,a_{n-1}$ with each letter repeated exactly thrice. Moreover, if a letter of the resulting word when read from the left occurs from the first time, then due to the

above construction, no larger letter could have occurred already twice. Likewise, if a letter occurs from the second time, again no larger letter could have occurred already thrice. Therefore, the resulting word satisfies the property from Definition 1.

Clearly the above construction can be reversed. Thus, the resulting map constitutes a bijection. □



(a) Network encoded by the word $aa|abb|bcc|c$.
(b) Network encoded by the word $abcabc|a|b|c$.
(c) Network encoded by the word $cbaabc|a|b|c$.
(d) Network encoded by the word $dcaa|abb|bc|cd|d$.

Figure 2: Maximally reticulated tree-child networks.

Fuchs et al. [13] made use of these words as an auxiliary tool to determine the cardinality of the $|\mathcal{TC}_{n,n-1}|$ subclass, and to discern its asymptotic behavior. However, they are important in their own right because statistical properties of the words have a direct reflection on the topology of the network. Consider for instance the mean distance between equal letters and their dispersion. The lowest possible value is zero and it corresponds to words of the type $aaabbbccc\ldots$ and their corresponding network, Figure 2a, is very regular. The maximum mean value is equal to $n-1$ and is achieved by words of the type $\alpha_{p_1}\alpha_{p_2}\cdots\alpha_{p_n}abc\cdots\alpha_n abc\cdots\alpha_n$, where $(p_1, p_2, \ldots, p_n)$ is any permutation of $\{1, 2, \ldots, n\}$, and the dispersion ranges from zero for the identity permutation, to a maximun value of $\sqrt{(n^2-1)/6}$ (reached when the permutation is $(n, n-1, \ldots, 1)$). These two cases correspond to Figures 2b and 2c, respectively. Words having the maximum dispersion are of the form $zy\cdots rq\,aaabbb\cdots ppp\,qqrr\cdots zz$, where we assumed that z is the n^{th} letter of the alphabet and p is the letter located in the $n-s$ alphabet's position. The length s of the prefix $zy\cdots q$ yielding a maximum dispersion depends on n, going asymptotically to a value $s = (4n-2)/9$ and implying a linear increase of the maximum dispersion. The concomitant value is $\sqrt{710}/27\,(n - 967/1420)$, and they have a mean value of $14/27(n-1)$, about half of the absolute maximum. For $n = 4$ the specific word is $dcaaabbbccdd$, and the highly irregular TCN associated to it is shown in Figure 2d.

2.2 Phylogenetic trees

Now we move to the other extreme case: TCNs with no reticulation nodes, i.e. *phylogenetic trees*. Next we show that there is a bijective relationship between phylogenetic trees with n leaves and words over an alphabet of $n-1$ letters satisfying the same conditions as in Definition 1. In this case letters are now repeated exactly twice instead of thrice. This is stated in the following proposition:

Proposition 3. *The set of phylogenetic trees on $[n]$ taxa is bijectively related to the class of words $\mathcal{B}_{n-1}$ built from a $(n-1)$-ary alphabet, so that in each word w every letter is repeated exactly twice, and for every prefix z of w we have $\#(z, a_i) = 0$ or $\#(z, a_i) \geq \#(z, a_j)$ for all $j > i$, and the function $\#(z, a_i)$ counts the occurrences of the i-th letter in z.*

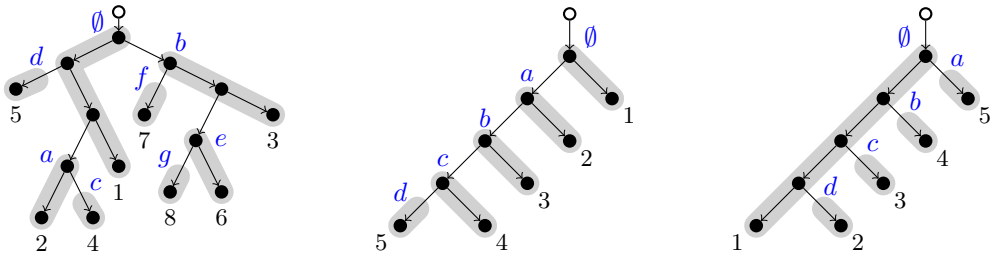
Proof. We shall give the bijection. Suppose that the leaves are labeled with elements of $\{1, 2, \ldots, n\}$. First, let us make the assignements $\{2 \to a, 3 \to b, 4 \to c, \ldots\}$. Next, index the path components of the tree in the following way: assign index 0 to the path from the root to the leaf labelled with number 1. For every node of the path, different from the root and the leaf, consider the path from that node to the descendant leaf with the lowest label and index that path according to the leaf's label and former assignment. Since any two members of the (root) path can not have common descendants not belonging to the path, it does not matter the order in which the indexing is done.

Continue this until all path-components are indexed, which will eventually happen because our trees are assumed to be connected.

Finally, we read the labels of each path-component starting with the 0-th one, follow with the a path writing firstly the name of the path followed by the name of all paths departing from it. Then proceed lexicographically until we reach the last one (see Figure 3 for some examples).

The resulting word uses $n-1$ letters, $a_1, \ldots, a_{n-1}$ with each letter repeated exactly twice. Moreover, if a letter of the resulting word when read from the left occurs from the first time, then due to the above construction, no longer letter could have occurred already twice.

Finally, it is straightforward to see that the above construction can be reversed. Thus, the resulting map is again a bijection. □



(a) Tree encoded by the word $bdaacbfecdegfg$.

(b) Tree encoded by the word $aabbccdd$.

(c) Tree encoded by the word $dcbaabcd$.

Figure 3: Examples of Tree encodings.

The last bijection provides an easy encoding of phylogenetic trees making their comparison straightforward. However, the codification is not a *succinct representation* because it uses twice the number of strictly necessary bits. This fact is seen by considering the leading term in the asymptotic expansion of the number $(2n-3)!!$ of phylogenetic trees, which is $\sqrt{2}\left(\frac{2n-2}{e}\right)^{n-1}$. The number of bits (obtained by taking the base 2 logarithm of the previous expression) goes as $(n-1)\log_2(n-1)$. On the other hand, our words contain $2(n-1)$ letters, and for each letter one shall require $\log_2(n-1)$ bits. Still, the proposed encoding is not much larger than Netwick's standard codification, which only uses n numbers but it also requires $2n-2$ parenthesis and $n-1$ commas. The important fact here is that Netwick's encoding is not a bijection.

Similarly as done for maximally reticulated networks, statistical analysis of the words can be performed. The very important difference is that in the former case words did not depend on the labels of the leaves, thus they only affect the topology of the network. But for phylogenetic trees, words depend not only on the topology but also on the labelling. Popular statistical indices for trees, such as the Collness or Sackin indices do not depend on the labelling, reflecting the structure of the tree. Therefore taking into account leaves' names in the statistical indices can only be useful to compare trees with the same names for the leaves, or to be significative if some non arbitrary or significant order can be stablished between the set of leaves, which represent the so-called *extant species*. One such order could be given by the population of the extant species. Regarding as before the distance between equal letters and considering the species tagged with label 1 to be the less populated, words with the minimum average distance (actually zero) $aabbcc\ldots$ correspond to trees where least populated species are systematically located closest to the root, as depicted in Fig. 3b. On the other hand, words with the largest mean distance, specifically $n-1$, correspond as before to trees with the maximal depth, but now with the least populated species located as far as possible to the root. They are all words of the form $\alpha_{p_1}\alpha_{p_2}\cdots\alpha_{p_n}abc\cdots\alpha_n$, where $(p_1, p_2, \ldots, p_n)$ is any permutation of $\{1, 2, \ldots, n\}$, and the dispersion ranges from zero for the identity permutation, to a maximun value of $\sqrt{(n^2-1)/3}$ reached when the permutation is $(n, n-1, \ldots, 1)$. This latter case is represented in Figure 3c. Words having the maximum dispersion are of the form $zy\cdots rq\,aabb\cdots pp\,qr\cdots yz$, where we assumed that z is the n^{th} letter of the alphabet and p is the letter located in the $n-s$-th alphabet position. The length s of the prefix $zy\cdots q$ yielding a maximum dispersion depends on n, going asymptotically to the value $s = (2n-1)/6$. The former fact implies a linear increase of the maximum dispersion $\sqrt{51}/9\,(n-32/51)$, having a mean value of $5/9(n-1)$.

As remarked by Diaconis and Holmes [7], combinatorialists often seek ways of walking through the space of all objects in a step-by-step way, useful for example to evaluate phylogenetic algorithms. It

is easy to generate the sequence of all this particular sets of words with minimal changes between any consecutive words. It translates in generating phylogenetic trees using minimal changes.

The following algorithm generates all $2n$ tuples $w = c_1 c_2 \dots c_{2n}$ of the n numbers $\{1, 2, \dots, n\}$, all numbers repeated twice, and satisfying the conditions stated in Proposition 3. An auxiliary vector of parities $d_2 d_3 \cdots d_n$ is used to reproduce the mirror symmetry.

T1. [**Initialize.**] Set $c_{2j-1} \leftarrow j$ and $c_{2j} \leftarrow j$ for $1 \leq j \leq n$, and set $d_j \leftarrow 0$ for $2 \leq j \leq n$. Also set $i \leftarrow 2$.

T2. [**Visit.**] Visit the word $w = c_1 c_2 \dots c_{2n}$.

T3. [**Locate left most.**] Set $j \leftarrow 1$ and repeat $j \leftarrow j + 1$ until $c_j = i$. Set $a \leftarrow c_j$.

T4. [**Direction?**] If $d_i = 0$ go to T5, otherwise ($d_i = 1$) go to T6.

T5. [**Move left.**] Set $k \leftarrow j - 1$ and repeat $k \leftarrow k - 1$ until $c_k < c_j$. If $k > 0$ go to 7. On the contrary, set $d_i \leftarrow 1$ and $i \leftarrow i + 1$. Terminate if $i = n + 1$, otherwise go to T3.

T6. [**Move right.**] Set $k \leftarrow j + 1$ and repeat $k \leftarrow k + 1$ until $c_k \leq c_j$. If $c_k \neq c_j$ go to 7. On the contrary, set $d_i \leftarrow 0$, $i \leftarrow i + 1$ and go to T3.

T7. [**Exchange.**] Set $c_j \leftarrow c_k$ and $c_k \leftarrow a$. Also set $i \leftarrow 2$ and go to T2.

The output sequence for $n = 3$ (corresponding to trees with 4 leaves) is

1	1	2	2	3	3
1	2	1	2	3	3
2	1	1	2	3	3
2	1	1	3	2	3
1	2	1	3	2	3

1	1	2	3	2	3
1	1	3	2	2	3
1	2	3	1	2	3
2	1	3	1	2	3
2	3	1	1	2	3

1	3	2	1	2	3
1	3	1	2	2	3
3	1	1	2	2	3
3	1	2	1	2	3
3	2	1	1	2	3

2.3 The conjecture

We conjecture there exists a *"natural bijection"* from every subclass $\mathcal{TC}_{n,k}$ of tree-child networks with n leaves and k reticulation nodes onto a newly defined class of words $\mathcal{C}_{n-1,k}$ which contains words with $n - 1 - k$ letters repeated twice and k letters repeated thrice, and satisfying similar conditions than the ones stated in Definition 1. More specifically, for any subclass of TCNs can be established a map

$$\psi \ : \ \mathcal{TC}_{n,k} \ \longrightarrow \ \mathcal{C}_{n-1,k} \ ,$$

with the cardinalities of the *domain* and *codomain* sets related by

$$|\mathcal{TC}_{n,k}| = \frac{n!}{(n-k)!} \, |\mathcal{C}_{n-1,k}| \, . \tag{2}$$

We identify the relating factor as the number of possible *injective functions* that can be built from a set of k elements into a set of n elements.

Strong evidence indicates that the conjecture shall be true. By solving the counting problem concerning $\mathcal{C}_{n,k}$, the numbers of tree-child networks predicted by the hypothesis (2) exactly coincide with all entries of the table provided by Cardona and Zhang [5] which cover all numbers of TCN subclasses up to eigth leaves. Furthermore simple analytic expressions can be extracted from the hypothesis, which coincide with already proven formulas for low numbers of reticulation nodes deduced by different methods: case $k = 1$ solved by L. Zhang in Ref. [23], case $k = 2$ proved by Cardona and Zhang [5] and $k = 3$ provided by Fuchs et al. [11]. Of course it also coincides with the extreme cases $k = 0$ and $k = n - 1$. Incidentally, due to a theorem provided in Ref. [5], it also coincides with the case $k = n - 2$.

Retrieving a precise relating function will lead to useful consequences. For example it will make possible a convenient encoding of TCNs, facilitating their comparison. The words will have to be equipped with extra characters to convert it in a bijection. For instance, in the case $k = n - 1$ the word represented in Fig. 2d will need to be completed as *dcaaa*3*bbb*1*cc*5*dd*2, where the label of the final leaf of every path is written after the third appearance of the corresponding letter.

In the next section the new class of words is defined and several counting formulas are obtained, as well as a precise asymptotic result.

3 Counting words

Let us consider the *class* $\mathcal{C}_{n,k}$ *of words built from a* n*-ary alphabet, where all words have the same length,* $2n+k$*. Moreover every word* w *has* k *letters repeated thrice,* $n-k$ *letters occur twice and for every prefix* z *of* w *we have* $\#(z,a_i)=0$ *or* $\#(z,a_i)\geq\#(z,a_j)$ *for all* $j>i$*. The function* $\#(z,a_i)$ *counts the occurrences of the* i*-th letter in* z*.*

Proposition 4. *The number of such words,* $c_{n,k}\equiv|\mathcal{C}_{n,k}|$*, satisfies the recurrence*

$$c_{n,k}=c_{n,k-1}+(2n+k-1)\,c_{n-1,k}\qquad \textit{with}\quad c_{0,0}=1\quad \textit{and}\quad k\in\{0,\ldots,n\}\,. \tag{3}$$

Proof. For the sake of clarity, we shall first consider the case $k=0$. In this situation all letters are repeated twice.

$$\begin{aligned}
c_{0,0} &= 1 \;:\; \emptyset \quad \text{(The empty word.)}\\
c_{1,0} &= 1 \;:\; \{aa\}.\\
c_{2,0} &= 3 \;:\; \{aabb, abab, baab\}.\\
c_{3,0} &= 15 \;:\; \{aabbcc, ababcc, baabcc, aabcbc, abacbc, baacbc, aacbbc,\\
&\qquad\qquad abcabc, bacabc, acabbc, acbabc, bcaabc, caabbc, cababc, cbaabc\}.
\end{aligned}$$

Numbers $c_{n,0}$ are just $(2n-1)!!$. Given the set corresponding to an alphabet of $n-1$ letters, in order to obtain the set associated with n letters one shall add to each word two repetitions for the new letter (the last one in the alphabet). One of them must necessarily go at the end, and the other one may go elsewhere, thus having $2n-1$ possibilities. From here onwards one shall define $c_{n,-1}=0$.

Moving to the general case, notice that words belonging to $\mathcal{C}_{n,k-1}$ can only contain the first $k-1$ letters of the alphabet repeated thrice. In order to retrieve elements of $\mathcal{C}_{n,k}$, one shall add the $k-$th letter of the alphabet to each word of the set $\mathcal{C}_{n,k-1}$. This letter can always be placed at the end of the word. It accounts for the first term on the right hand side of Eq. (3). Done in this fashion, in order to complete the set $\mathcal{C}_{n,k}$, there only remain the words ending with the last letter of the alphabet. These words can be obtained from the set $\mathcal{C}_{n-1,k}$ by adding a new letter to its alphabet, that is, two repetitions for the $n-$th letter. One of them has to necessarily go at the end, but the other one can be placed in any of the $2n+k-1$ other positions. It accounts for the second term on the right hand side of Eq. (3). Incidentally, in order to obtain the set $\mathcal{C}_{n,n}$, the fist procedure already generates all words with three repetitions, and it places the last $n-$th letter at the end of each word. Hence the convenience to define $c_{i,i+1}=0$. □

Recurrence (3) is able to determine all $c_{n,k}$ numbers. The lowest values are

$$\begin{array}{ccccccc}
 & & & 1 & & & \\
 & & & 1 & 1 & & \\
 & & & 3 & 7 & 7 & \\
 & & 15 & 57 & 106 & 106 & \\
 & 105 & 561 & 1515 & 2575 & 2575 & \\
945 & 6555 & 23220 & 54120 & 87595 & 87595 &
\end{array} \tag{4}$$

where rows and columns correspond to n and k, respectively.

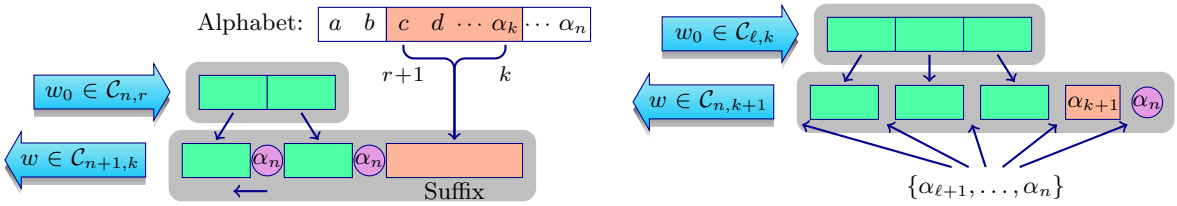
Proposition 5. *Numbers* $c_{n,k}$ *satisfy the following recurrence equations*

$$c_{n+1,k} \;=\; \sum_{r=0}^{k}(2n+r+1)\,c_{n,r}\,, \tag{5a}$$

$$c_{n,k+1} \;=\; \sum_{\ell=k+1}^{n}\prod_{i=\ell+1}^{n}(2i+k)\;c_{\ell,k}\,. \tag{5b}$$

Proof. Both equations follow by considering Eq. (3) as a first order recurrence equation, $y_{m+1}=a_m y_m+b_m$, and hence applying its standard solution, see Section B of the Supplementary Information (SI). To obtain Eq. (5a), the independent term b_m must be identified with $c_{n,k-1}$, solve for n and after that, shift n by one unit. Whereas to obtain Eq. (5b) the independent term has to be identified with $(2n+k-1)c_{n-1,k}$, and to solve for k. □

Every counting equation implicitly describes a recursive procedure to generate the entire class. Formula (5a) implies that all words composing $\mathcal{C}_{n+1,k}$ can be obtained by adding up disjoint sets, one for each member of the sum (5a). All words belonging to one such set share a common suffix formed by $k-r$ ordered letters of the alphabet, from the $(r+1)$-th to the k-th letter. Different prefixes are built from words in $\mathcal{C}_{n,r}$, adding the $(n+1)$-th letter of the alphabet twice, one at the end of the word and the other one inserted in all possible positions. See diagram (a) shown in Figure 4.



(a) Generation associated to formula (5a).

(b) Generation associated to formula (5b).

Figure 4: Methods to recursively generate the classes of words.

Regarding the other expression, Eq. (5b), the members of an arbitrary disjoint set are obtained from words belonging to $C_{\ell,k}$ by first adding the $(k+1)$-th letter of the alphabet at the end of that word. Then add two repetitions of the $(\ell+1)$-th letter, one at the end and the other in all possible positions. After that add similarly the $(\ell+2)$-th letter, one placed at the end and the other one elsewhere. Continue, in the same vein, until placing the last letter of the alphabet. The method is depicted in the diagram (b) displayed in Figure 4 .

Next an alternative formula is provided. It is useful to obtain explicit expressions for low numbers of k.

Proposition 6. *The number of words contained in $\mathcal{C}_{n,k}$ is given by*

$$c_{n,k} = \sum_{i=0}^{k} \frac{(2n+2k-i-1)!!}{(k-i)!}\, a_i\,, \tag{6}$$

where the coefficients a_i are determined by the recursive equation

$$a_i = -\sum_{j=1}^{i} \frac{(3i-3+j)!!}{(3i-3)!!}\,\frac{a_{i-j}}{j!}\,, \qquad \text{and} \quad a_0 = 1. \tag{7}$$

The first terms of the a_i sequence are $\{1, -1, \frac{1}{6}, \frac{17}{48}, -\frac{283}{1512}, -\frac{467}{9216}, \frac{66329}{1297296}, \frac{8915}{4644864}, \ldots\}$.

The proof follows easily by induction. It can be found in the Section A of the SI file.

3.1 Enumeration

Given a combinatorial class, an important issue is to being able to enumerate all its elements. Next we provide an algorithm, based on the recurrence (5a), which sequentially generates all words that belong to $\mathcal{C}_{n,k}$. Specifically it generates all $m = 2n+k$ tuples $w = c_1 c_2 \ldots c_m$ of the n numbers $\{1, 2, \ldots, n\}$ with k numbers repeated thrice, and the remaining $n-k$ numbers repeated twice. They do so by satisfying the conditions stated at the beginning this section. An auxiliary vector $a_1 a_2 \cdots a_n$ is used to track the number of available interchanges.

A1. [**Initialize.**] Set $c_{2i-1} \leftarrow i$ and $c_{2i} \leftarrow i$ for $1 \le i \le n$, also set $c_{2n+i} \leftarrow i$ for $1 \le i \le k$. Set $a_i \leftarrow 0$ for $1 \le i < n$ and set $a_n \leftarrow \min(k, n-1)$.

A2. [**Visit.**] Visit the word $w = c_1 c_2 \ldots c_m$ and set $j \leftarrow 2$.

A3. [**Try to move left.**] Find least p such that $c_p = j$. Find greatest $q < p$ such that $c_q < j$. If q is found set $c_p \leftarrow c_q$ and $c_q \leftarrow j$. Also set $a_i \leftarrow 0$ for $1 \le i < j-1$ and go to A2.

A4. [**Try to move right.**] Find least $q > p$ such that $c_q = j$. Find least $s > q$ such that $c_s < j$ $(s < n+q)$. If found and $c_s \le a_j$ then set $c_p \leftarrow c_s$, $c_s \leftarrow j$ and actualize the auxiliary vector: set $a_{j-1} \leftarrow \min(j-1, a_{j-1}+1)$ and $a_i \leftarrow 0$ for $1 \le i < j-1$ and go to A6.

A5. [**Increase j.**] Set $j \leftarrow j+1$. Terminate if $j > n$, otherwise set $out \leftarrow false$.

A6. [**Prepare to actualize.**] Set $\ell \leftarrow j-1$ and $r \leftarrow a_\ell$. Also set $u \leftarrow \min(\ell, r)$, $t \leftarrow 1$, $s \leftarrow 0$ and $p \leftarrow 1$.

A7. [**Actualize.**] Repeat $p \leftarrow p+1$ until $c_p \leq \ell$. If $t \leq \ell$ set $c_p \leftarrow t$, otherwise set $c_p \leftarrow t-\ell$. If $s = 1$ and $r = \ell + u$ we are done; go to A8. If $s = 1$ then set $t \leftarrow t+1$. If $r \leq \ell$ set $s \leftarrow 1-s$. Return to A7.

A8. [**Visit?**] If $out = true$ then go to A2, otherwise set $out \leftarrow true$ and go to A3.

The output is sorted following the examples of words displayed after *Definition* 1, case $k = n-1$, or the words following *Proposition* 4, case $k = 0$.

The last algorithm is a bit tricky. It turns out that it is much simpler to give an explicit enumeration. Equipped with the table of cardinalities $c_{n,k}$ it is very easy to recursively determine in which position a given word has been generated. This task is done by the following algorithm, the generic input being any arbitrary word of the type $w = c_1 c_2 \dots c_m \in \mathcal{C}_{n,k}$.

E1. [**Initialize.**] Determine n, the number of distinct characters. Also determine k, the number of characters repeated thrice. Set $P \leftarrow 1$, the position of the word in the list.

E2. [**Easy case?**] If $k = n$ remove the last character of the word (it is necessarily α_n). And set $k \leftarrow k-1$.

E3. [**Localize last char.**] Set $p \leftarrow 2n+k$ and repeat $p \leftarrow p-1$ until $c_p = \alpha_n$.
Set $q \leftarrow p-1$ and repeat $q \leftarrow q-1$ until $c_q = \alpha_n$.

E4. [**Actualize.**] Remove last $2n+k-p$ characters and the other repetition of α_n (located at position q). Set $k \leftarrow p-2n$ and also set $n \leftarrow n-1$.

E5. [**Accumulate.**] Set $P \leftarrow P + (p-q-1)\, c_{n,k} + \sum_{r=0}^{k-1} (2n+r+1)\, c_{n,r}$. Terminate if $n = 0$, otherwise go to E2.

It is of course possible to reverse last algorithm. That is, specifying n, k and an integer $1 \leq P \leq c_{n,k}$, to determine to which word it corresponds. Such algorithm could be useful to generate random words.

3.2 Asymptotic behavior

We provide an asymptotic expression for $c_{n,k}$ valid for small k, more specifically for $k < \sqrt{n}$.

Proposition 7. *For $k < \sqrt{n}$, numbers $c_{n,k}$ grow as*

$$c_{n,k} = \sqrt{2}\,\frac{e^{-n}\,(2n)^{n+k}}{k!}\left\{1 - \sqrt{\frac{\pi}{2}}\,k\,(2n)^{-1/2} + \frac{14k^2-2k-1}{12}\,(2n)^{-1} - \sqrt{\frac{\pi}{2}}\,k\,\frac{31k^2+3k-26}{48}\,(2n)^{-3/2} + \frac{2900k^4-264k^3-10016k^2+6876k+21}{6048}\,(2n)^{-2} + \mathcal{O}\left(\frac{k^2}{n}\right)^{5/2}\right\} \quad (8)$$

The starting point to obtain this expression is Eq. (6). In this regime of n and k values, it can be considered the expansion of the numerator $(2n+2k-i-1)!!$ for big values of n. Specifying the series for our particular case, a factor involving the power $n^{-i/2}$ appears, indicating that only the first terms of the sum (6) are relevant for this regime of parameters. The detailed proof can be found in the Section C of the SI file. More correction terms are easily computable as can be seen there.

4 Implications of the conjecture

It is plain that it would be desirable to establish a bijection between the subclass of tree-child networks $\mathcal{TC}_{n,k}$ and the class of words $\mathcal{C}_{n,k}$. Such bijection would provide a proper codification for the networks that would greatly help in their comparison, the enumeration procedures provided in section 3.1 could be used, but it would specially help to study the combinatorial and stochastic properties of those networks, linking them to the properties of words. Summing up, the bijection would greatly help to

characterize a "typical network". But usually it is not so simple to design a bijection. It is often much simpler to count directly the elements of a set rather than to provide a bijection with a different set whose counting is known. In order to fully prove our conjecture (2) it may be easier to do so probably via some inductive argument, by relating the cardinals of subclasses of networks. Thus, given the main conjecture (2) and the recurrence between words (4), we state the following proposition:

Proposition 8. *The cardinalities* $|\mathcal{TC}_{n,k}|$ *satisfy*

$$(n-k)|\mathcal{TC}_{n,k}| = (n+1-k)(n-k)|\mathcal{TC}_{n,k-1}| \; + \; n(2n+k-3)|\mathcal{TC}_{n-1,k}|\,, \tag{9}$$

with initial values $|\mathcal{TC}_{1,0}| = 1$ *and* $|\mathcal{TC}_{i,-1}| = |\mathcal{TC}_{i,i}| = 0 \;\; \forall \; i\,.$

This recurrence exactly reproduces the table given in Ref. [5]. Table 1, placed in the SI file displays the counts of TCNs up to 11 leaves and all possible reticulation numbers.

Similarly, simply adding the falling factorial to Eq. (5a), the following result is obtained.

Proposition 9. *The following relation also holds:*

$$(n-k)!\,|\mathcal{TC}_{n,k}| = n\sum_{r=0}^{k}(2n+r-3)\,(n-1-r)!\,|\mathcal{TC}_{n-1,r}|\,. \tag{10}$$

Now, the inductive argument would consist of adding reticulations nodes starting from the initial set of phylogenetic trees with $n-1$ leaves or, alternatively, to begin with an arbitrary TCN and to start deleting reticulations until reaching a tree.

The following equality follows similarly from Eq. (5b) by adding the falling factorial.

Proposition 10. *Yet another relation for* $|\mathcal{TC}_{n,k}|$*:*

$$|\mathcal{TC}_{k+m+1,k}| = \sum_{\ell=0}^{m}(\ell+2)\left[\prod_{i=\ell+1}^{m}\left(1+\frac{k}{i+1}\right)\Big(2i+3k-1\Big)\right]|\mathcal{TC}_{k+\ell+1,k-1}|\,, \tag{11}$$

valid for $k \geq 1\,.$

This last equation (11) relates the number of elements of $\mathcal{TC}_{n,k}$ with those of the networks with one less reticulation, as well as all possible number of leaves. In this case the inductive reasoning would involve deleting leaves till reaching a maximally reticulated TCN, a particular subclass already counted in Refs. [11, 13]. Since we believe that this is a promising way to prove the conjecture, Eq. (2), particular instances of this recurrence for low values of m are displayed bellow.

$$|\mathcal{TC}_{k+1,\,k}| = 2\,|\mathcal{TC}_{k+1,\,k-1}| \tag{12a}$$

$$|\mathcal{TC}_{k+2,\,k}| = 3\,|\mathcal{TC}_{k+2,\,k-1}| + \Big(1+\frac{k}{2}\Big)(3k+1)\,2\,|\mathcal{TC}_{k+1,\,k-1}| \tag{12b}$$

$$|\mathcal{TC}_{k+3,\,k}| = 4\,|\mathcal{TC}_{k+3,\,k-1}| + \Big(1+\frac{k}{3}\Big)(3k+3)\Big(3\,|\mathcal{TC}_{k+2,\,k-1}| + \Big(1+\frac{k}{2}\Big)(3k+1)\,2\,|\mathcal{TC}_{k+1,\,k-1}|\Big) \tag{12c}$$

Let us notice how, in this fashion, we recover the principal recurrence (9):

$$|\mathcal{TC}_{k+m+1,k}| = (m+2)|\mathcal{TC}_{k+m+1,k-1}| \; + \; \Big(1+\frac{k}{m+1}\Big)(3k+2m-1)|\mathcal{TC}_{k+m,k}|\,.$$

The first equation (12a) was already proven in Ref. [5] (Theorem 12 therein). We presume that proving the particular case (12b) is a decisive step towards the proof of the conjecture. Moreover we think that the correct interpretation of Eq. (11) could bring us to the desired bijection. We believe so because the letters repeated thrice in our words are the first of the alphabet, then a feasible strategy would consist in removing leaves, possibly in an ordered way, from a given TCN until reaching a maximally reticulated TCN, then setting the labels of the paths according to Fuchs bijection, Proposition 2, and after that reconstructing the network adding the previously removed leaves.

Relation (6) provides a straightforward method for obtaining explicit formulae for the number of TCNs with few reticulations. Let us rewrite it in terms of the new coefficients $b_i \equiv i!\,a_i$.

Proposition 11. *A final, convenient expression for $|\mathcal{TC}_{n,k}|$ is given in the following:*

$$|\mathcal{TC}_{n,k}| = \binom{n}{k} \sum_{i=0}^{k} \binom{k}{i} (2n+2k-i-3)!!\, b_i\,, \tag{13}$$

where the coefficients b_i are determined by the recursive equation

$$b_i = -\sum_{j=1}^{i} \binom{i}{j} \frac{(3i-3+j)!!}{(3i-3)!!}\, b_{i-j}\,, \qquad \textit{and} \quad b_0 = 1. \tag{14}$$

The first terms of the sequence are $\{1, -1, \frac{1}{3}, \frac{17}{8}, -\frac{283}{63}, -\frac{2335}{384}, \frac{331645}{9009}, \ldots\}$. Notice that Eq. (14) also follows from the condition $|\mathcal{TC}_{n,n}| = 0$.

The number of TCNs in closed form can be obtained analytically with the help of the previous relation (13). We shall list in the following some of them:

$$|\mathcal{TC}_{n,1}| = \binom{n}{1}\left\{(2n-1)!! - (2n-2)!!\right\} \tag{15a}$$

$$|\mathcal{TC}_{n,2}| = \binom{n}{2}\left\{(2n+1)!! - 2(2n)!! + \frac{1}{3}(2n-1)!!\right\} \tag{15b}$$

$$|\mathcal{TC}_{n,3}| = \binom{n}{3}\left\{(2n+3)!! - 3(2n+2)!! + (2n+1)!! + \frac{17}{8}(2n)!!\right\} \tag{15c}$$

$$|\mathcal{TC}_{n,4}| = \binom{n}{4}\left\{(2n+5)!! - 4(2n+4)!! + 2(2n+3)!! + \frac{17}{2}(2n+2)!! - \frac{283}{63}(2n+1)!!\right\} \tag{15d}$$

Last expression, $k = 4$, is completely new, whereas the previous ones agree and simplify the existing formulas in a compact manner. An equivalent expression to (15a) was first provided by L. Zhang [23], a direct formula for TCNs with two reticulation nodes was first given by Cardona and Zhang [5], while a closed formula for $|\mathcal{TC}_{n,3}|$ can be found in Ref. [11].

Coefficients b_i form a rather odd sequence, as can be grasped from Fig. 5. There it is plotted the ratio of two consecutive terms, Panel 5a. As can be seen this quantity oscillates for every increment of one unit of i, this means that b_i coefficients can be grouped in pairs of consecutive elements having the same sign. We have checked numerically that this happens almost all the time. Also, the overall tendency of $|b_i|$ seems to be bounded by a known expression, as shown in the right panel 5b. It appears as if $\frac{2}{i}\ln\left(\frac{|b_i|}{\Gamma(i/2)}\right) < 1\ \forall\, i$.

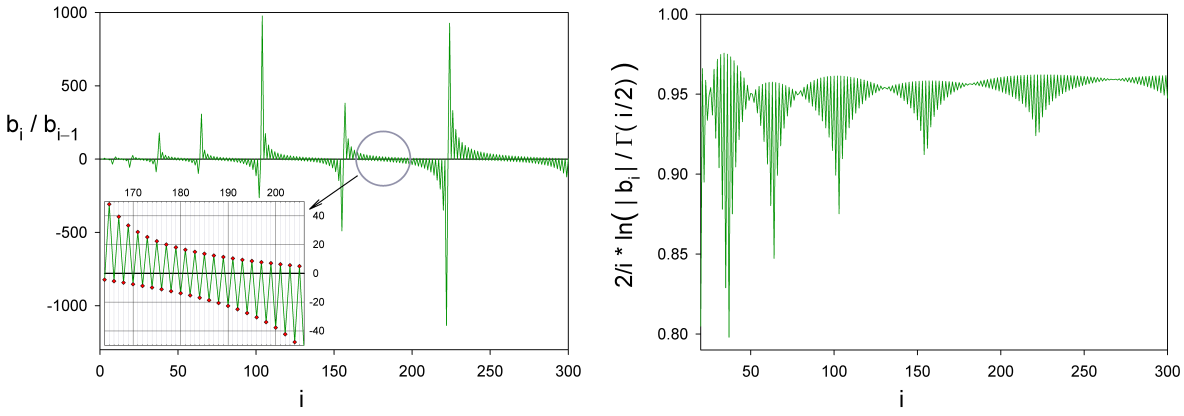
It is also instructive to study the numerical behavior of the terms composing the sum (13) for big n and several k values. In particular, in Fig. 6 we depict, for a fixed $n = 625$ and several k, the logarithm of the absolute value of all the terms forming the sum. In each case, the horizontal line represents the logarithm of the sum result. The first panel depicts the moduli for $(n = 625, k = 50)$, with $k < \sqrt{n}$; the second one, $(n = 625, k = 312)$, with $k \approx n/2$; finally, the third one depicts $(n = 625, k = 624)$, that is, $k = n - 1$. Bearing in mind that the first term in the sum (13) constitutes an absolute upper bound to the sum itself, great cancelations occur in the summation for each case until each curve reaches the horizontal line.

4.1 Asymptotic expression

The concomitant expression for $|\mathcal{TC}_{n,k}|$ is obtained following (8) with $n \to n-1$, and adding the factor $\frac{n!}{(n-k)!}$.

Proposition 12. *For $k < \sqrt{n}$, numbers $|\mathcal{TC}_{n,k}|$ grow as*

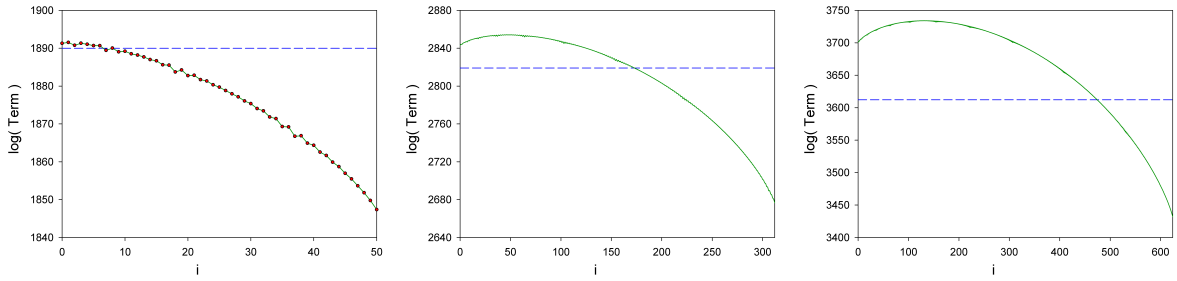
$$|\mathcal{TC}_{n,k}| \;=\; \binom{n}{k}\sqrt{2}\, e^{-n}\, (2n)^{n-1+k}\Bigg\{1 - \sqrt{\frac{\pi}{2}}\, k\, (2n)^{-1/2} + \frac{14k^2 - 26k + 11}{12}\, (2n)^{-1}$$
$$-\sqrt{\frac{\pi}{2}}\, k\, \frac{31k^2 - 93k + 70}{48}\, (2n)^{-3/2} + \frac{2900k^4 - 14376k^3 + 25264k^2 - 19332k + 5565}{6048}\, (2n)^{-2} + \mathcal{O}\left(\frac{k^2}{n}\right)^{5/2}\Bigg\} \tag{16}$$



(a) Ratio of consecutive terms

(b) Growth Rate

Figure 5: Behavior of the b_i coefficients. Most of the time, two consecutive terms possess the same sign, although few counterexample appears. This feature is clearly seen in the first plot (5a), where the ratio of consecutive coefficients oscillates for each i. To study the growth of coefficients b_i, the right plot (5b) depicts the quantity $\frac{2}{i}\ln\left(\frac{|b_i|}{\Gamma(i/2)}\right)$.



(a) n=625 and k=50

(b) n=625 and k=312

(c) n=625 and k=624

Figure 6: Decimal logarithm of the absolute value of the terms forming the sum (13), including the $\binom{n}{k}$ factor. Blue line corresponds to the logarithm of the final sum result.

Fuchs et al. [10] were able to reproduce the asymptotic expressions of $|\mathcal{TC}_{n,k}|$ for large n and $k = 1, 2, 3$ by employing a rather involved method based on generating functions. We recover the (few) previously known expansions and extend them to any k and n, along with several additional corrections to the leading term.

In the Figures 7 and 8, placed in the SI file, we depict the accuracy of the asymtotic expansion (16), at different orders of approximation, for two particular cases, $k = 4$ and $k = 5$, as a function of n. Usually, but not always, successive approximations overestimate and underestimate the exact value.

5 Conclusions

We have shown in the present contribution that employing words for counting tree-child networks is extremely useful, providing simple formulas for exactly counting TCNs with n leaves and k reticulation nodes. The main result is provided by a recursive relation, from which any value for $|\mathcal{TC}_{n,k}|$ can be easily obtained. What was considered to date a difficult mathematical problem is now tractable in a simple and compact manner.

Additionally, closed formulas for small number of reticulations k and arbitrary n leaves are also given explicitly. The corresponding particular analytic forms are far more compact than previous results known in the literature. Furthermore, an asymptotic expression is also provided.

We are aware that our results ultimately rely on a conjecture, but we provide different arguments

to support it. We believe that this conjecture will constitute an important and useful tool, as well as a challenging problem to solve for the Phylogeny's community. Since the enumeration and probabilistic study of the introduced words is simple, by proving the main conjecture, and from the results presented in this work, relevant advances in the understanding of TCNs will certainly be achieved in the near future. Among them, we shall focus our attention on the simple enumeration of networks or their practical encoding and a better combinatorial and stochastic characterization.

Acknowledgements

We thank Prof. Mike Steel for his practical recommendations and useful comments. J. Batle acknowledges fruitful discussions with J. Rosselló, Maria del Mar Batle, Regina Batle, and Maria Vallespir-Socias. The authors received no funding for the present research.

Competing interests

The authors declare no competing interests.

Author contributions

MP: Conceptualization and Algorithm development. MP and JB: Study design; Investigation; Formal analysis; Visualization. Writing – original draft and Review.

† *Miquel Pons. *E-mail address*: m.pons-viver@uib.es
†Josep Batle. *E-mail address*: jbv276@uib.es

† Departament de Física, Universitat de les Illes Balears, 07122 Palma de Mallorca, Balearic Islands, Spain.
* *Corresponding author.*

References

[1] M. Bienvenu, A. Lambert, and M. Steel. Combinatorial and stochastic properties of ranked tree-child networks. *arXiv:2007.09701 [math.PR]*, 2021.

[2] Magnus Bordewich and Charles Semple. Determining phylogenetic networks from inter-taxa distances. *J. Math. Biol.*, 73(2):283–303, 2016.

[3] M. Bouvel, P. Gambette, and M. Mansouri. Counting phylogenetic networks of level 1 and level 2. *arXiv:1909.10460 [math.CO]*, 2019.

[4] G. Cardona, F. Rossello, and G. Valiente. Comparison of tree-child phylogenetic networks. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 6(4):552–569, 2009.

[5] G. Cardona and L. Zhang. Counting tree-child networks and their subclasses. *J. Comput. Syst. Sci.*, 114:84–104, 2020.

[6] Zhi-Zhong Chen and Lusheng Wang. Algorithms for reticulate networks of multiple phylogenetic trees. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 9(2):372–384, 2012.

[7] Persi Diaconis and Susan Holmes. Matchings and phylogenetic trees. *Proc. Natl. Acad. Sci.*, 95(14600–14602), 1998.

[8] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

[9] Peter Forster, Lucy Forster, Colin Renfrew, and Michael Forster. Phylogenetic network analysis of SARS-CoV-2 genomes. *Proc. Natl. Acad. Sci.*, 117(17):9241–9243, 2020.

[10] M. Fuchs, B. Gittenberger, and M. Mansouri. Counting phylogenetic networks with few reticulation vertices: tree-child and normal networks. *Australas. J. Combin.*, 73(2):385–423, 2019.

[11] M. Fuchs, B. Gittenberger, and M. Mansouri. Counting phylogenetic networks with few reticulation vertices: Exact enumeration and corrections. *arXiv:2006.15784 [math.CO]*, 2021.

[12] Michael Fuchs, En-Yu Huang, and Guan-Ru Yu. Counting phylogenetic networks with few reticulation vertices: A second approach. *arXiv:2104.07842 [math.CO]*, 2021.

[13] Michael Fuchs, Guan-Ru Yu, and Louxin Zhang. On the asymptotic growth of the number of tree-child networks. *European Journal of Combinatorics*, 93:103278, 2021.

[14] A. D. Gunawan, H. Yan, and L. Zhang. Compression of phylogenetic networks and algorithm for the tree containment problem. *J. Comput. Biol.*, 26(3):285–294, 2019.

[15] A. D. M. Gunawan, J. Rathin, and L. Zhang. Counting and enumerating galled networks. *Discrete Appl. Math.*, 283:644–654, 2020.

[16] Daniel H. Huson. Tutorial: Introduction to phylogenetic networks. Technical report, Center for Bioinformatics, Tübingen University, 2006.

[17] Daniel H. Huson, Regula Rupp, and Celine Scornavacca. Phylogenetic networks: Concepts, algorithms and applications. *Systematic Biology*, 61(1):174–175, 2011.

[18] Daniel H. Huson and Celine Scornavacca. A survey of combinatorial methods for phylogenetic networks. *Genome Biology and Evolution*, 3(1):23–35, 210.

[19] C. McDiarmid, C. Semple, and D. Welsh. Counting phylogenetic networks. *Ann. Comb.*, 19(1):205–224, 2015.

[20] Richard P. Stanley. *Enumerative Combinatorics*, volume 2. Cambridge University Press, 1999.

[21] Leo van Lersel, Steven Kelk, Regula Rupp, and Daniel Huson. Phylogenetic networks do not need to be complex: using fewer reticulations to represent conflicting clusters. *Bioinformatics*, 26(12):i124–i131, 2010.

[22] L. Zhang. Clusters, trees and phylogenetic network classes. In T. Warnow, editor, *Bioinformatics and Phylogenetics –Seminal Contributions of Bernard Moret*, volume 29, pages 277–315. Springer Nature, 2019.

[23] L. Zhang. Generating normal networks via leaf insertion and nearest neighbor interchange. *BMC Bioinformatics*, 20:642, 2019.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInformationPonsBatle.pdf