Asymptotic Enumeration of Compacted Binary Trees of Bounded Right Height*

Antoine Genitrini† Bernhard Gittenberger‡ Manuel Kauers§
Michael Wallner‡

September 11, 2019

Abstract

A compacted binary tree is a graph created from a binary tree such that repeatedly occurring subtrees in the original tree are represented by pointers to existing ones, and hence every subtree is unique. Such representations form a special class of directed acyclic graphs. We are interested in the asymptotic number of compacted trees of given size, where the size of a compacted tree is given by the number of its internal nodes. Due to its superexponential growth this problem poses many difficulties. Therefore we restrict our investigations to compacted trees of bounded right height, which is the maximal number of edges going to the right on any path from the root to a leaf.

We solve the asymptotic counting problem for this class as well as a closely related, further simplified class.

For this purpose, we develop a calculus on exponential generating functions for compacted trees of bounded right height and for relaxed trees of bounded right height, which differ from compacted trees by dropping the above described uniqueness condition. This enables us to derive a recursively defined sequence of differential equations for the exponential generating functions. The coefficients can then be determined by performing a singularity analysis of the solutions of these differential equations.

Our main results are the computation of the asymptotic numbers of relaxed as well as compacted trees of bounded right height and given size, when the size tends to infinity.

Keywords: Compacted trees, Enumeration, D-finiteness, Analytic Combinatorics, Directed Acyclic Graphs, Chebyshev Polynomials.

*This research was supported by the Austrian Science Fund (FWF) grant SFB F50-03.

†Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu 75005 Paris. Antoine.Genitrini@lip6.fr

‡Technische Universität Wien, Wiedner Hauptstraße 8-10/104, 1040 Wien, Austria. {Bernhard.Gittenberger,Michael.Wallner}@tuwien.ac.at

§Institute for Algebra, Johannes Kepler University, Altenberger Strasse 69, 4040 Linz, Austria. Manuel@Kauers.de

1 Introduction

Most trees contain redundant information in form of repeated occurrences of the same subtree[1]. In order to get an efficient representation in memory, these trees can be compacted by representing each occurrence only once. The removed subtrees are replaced by pointers which link to the shared subtree. Such structures are classically named as *directed acyclic graphs* or short as *DAGs*.

Flajolet *et al.*, in their extended abstract [15], analyzed in detail the gain in memory of the compaction. Some proofs have been omitted and have not been stated later. This gap was closed in [10], where the framework was extended to other DAG structures and analyzed in the context of XML compression. Furthermore, Ralaivaosaona and Wagner extended in [27] the analysis of the gained memory to simply generated families of trees.

The latter two papers on the quantitative analysis of the compaction process, studied the transformation of a given set of trees of given size to the set of compacted trees in order to determine the average rate of compaction. We focus on a different aspect, namely the enumeration problem of compacted binary trees. On the one hand, enumerating combinatorial structures is important if one wants to understand shape characteristics of large random structures or for uniform random generation of those structures. On the other hand, the enumeration of particular classes of DAGs is in general a difficult problem which requires the extension of combinatorial methodology and is therefore interesting in its own right.

One of the difficulties in the enumeration of compacted binary trees lies in the fact that a compacted binary tree of size n could arise from a binary tree whose size belongs to the whole interval $n, \ldots, 2^n$. Thus, a brute-force approach is hopeless.

The first papers about the enumeration of DAGs appeared in the 1970's. Robinson presented two distinct approaches [28,30] based either on some inclusion-exclusion method or on Pólya's enumeration theory [26]. Combining combinatorial arguments and analytic methods the asymptotic number of labeled DAGs was determined in [4], for connected structures then in [5]. The first investigation of shape parameters seems to go back to McKay [22]. Recently, enumeration results for many particular classes of DAGs can be found in the literature, see for instance [7–9, 16, 20, 21, 33–35], as well as investigations on the (random) generation of particular DAGs, see [3, 11, 23, 24].

A now classical way for enumeration is the use of generating functions. In this context, precisely for labeled structures (see paper [29]), Robinson designed generating functions of a very particular nature to solve an asymptotic counting problem concerning DAGs. The classical types of generating functions like ordinary and exponential ones were not suited for the problem.

We are facing the same problem in the enumeration of compacted trees. Indeed, due to the fact that compacted trees are unlabeled combinatorial structures, which are moreover closely related to plane trees, a treatment with ordinary generating functions will be the first choice. However, the fast growth of the counting sequence requires the use of exponential generating functions. In order to be able to get asymptotic results, we will confine ourselves to certain subclasses of the class of compacted trees as well as some related classes by relaxing

[1]In the rest of the paper, a subtree of a given tree contains a root and all its descendants in the original tree. Such a substructure is sometimes called a fringe subtree.

certain conditions. Moreover, we will develop a calculus for exponential generating functions designed for these classes. Bounding the right height of our DAGs leads to a sequence of D-finite functions (see [19, 32] for introductions to the subject) for which it is possible to analyze their differential equations and obtain finally our main result. Likewise, in other enumeration problems for particular classes of DAGs bounding a certain parameter turned intractable recurrences into D-finite ones. Examples are the enumeration of certain classes of lambda-terms [7–9] or increasing series-parallel DAGs [6].

Plan of the paper

Our combinatorial structures are based on the fundamental properties of the compaction procedure. We will first analyze some properties of this classical procedure (linked to the *common subexpression problem*) in Section 2.

Then we will define the basic concepts and state our main results in Section 3, see Theorems 3.3 and 3.4.

Some basic observations concerning the structure of compacted trees will then be presented in Section 4.

These will help us to state a combinatorial and (most importantly) recursive specification of the problem in Section 5. A further important result is the derivation of *a recurrence relation for the number of compacted binary trees*, see Theorem 5.1. This recurrence is not classical at all, and we are not able to solve it explicitly.

Due to this fact, we follow yet a different approach in the remaining part of this work: We will use exponential generating functions to model our problem, as the superexponential growth rate of the counting sequence suggests, though we are dealing with unlabeled combinatorial structures. Therefore, a new calculus translating certain set operations for classes of compacted trees into algebraic operations of exponential generating functions will be developed in Section 6.

Section 7 is devoted to a simplified problem, the study of the counting problem of *relaxed binary trees*. These DAGs are in a sense compacted trees where the restriction of uniqueness on the subtrees is relaxed. In particular, compacted trees are a subset of relaxed binary trees. With the same methods as used on compacted trees we are able to derive a recurrence relation. However, this recurrence relation is as difficult as the first one for compacted trees.

A natural constraint for compacted trees seems to bound some specific depth limit, the so-called right height. This is the maximal number of edges directed to the right which appear on any path from the root to a leaf. In Section 7, the calculus developed in Section 6 enables us to derive a differential equation for the generating function of relaxed trees for each bound k on the right height. This sequence of D-finite differential equations follows a rather explicit recursive scheme, presented in Theorem 7.10 which allows us to analyze the dominant singularities of the solutions of the differential equation for any k. Eventually, this strategy is successful and we are able to determine the asymptotic number of relaxed binary trees of bounded right height.

Finally, in Section 8 we modify the results of the previous section to cover the case of compacted trees as well. Again, we derive a sequence of D-finite differential equations where, as in Section 7, the dominant singularities of the generating function are regular singularities of the differential equation. This allows us to extract the asymptotic behavior of the counting sequence, which contains irrational powers of n. The necessary information is directly extracted from the differential equations. Except for the first few, they do not have closed-form solutions.

2 Creating a compacted tree

Many problems in computer science and computer algebra involve redundant information. A strategy to save memory is to store every instance only once and to point to already existing instances, whenever an instance appears repeatedly. In [15, Proposition 1] a compression algorithm was presented, and it was shown that for a given tree of size n, its compacted form can be computed in expected time $\mathcal{O}(n)$. However, such procedures have been known since the 1970's (see [13,15] and especially the "value-number method" in compiling [2, Section 6.1.2]). Figure 1 shows this procedure, which follows a top-down decomposition scheme (i.e. post-order traversal) of labeled binary trees. Every node (or actually the subtree whose root is the respective node) is associated with a "unique identifier" (*uid*). Two subtrees are equivalent if and only if the uid's are the same.

```
function UID(T : tree) :  integer;

global counter : integer, Table : list;
begin
   if  T  =  nil
      then  return(0);
      else
         triple  :=  <root(T),UID(left(T)),UID(right(T))>;
         if  Found(triple,Table)
            then  return(value_found);
            else  counter  :=  counter+1;
               Insert  pair  (triple,counter)  in  Table;
               return(counter);
         fi
   fi
end
```

Figure 1: The `UID` procedure from [15, Fig. 2] which computes "unique identifiers" for all (fringe) subtrees of a given binary tree T. It is assumed that `counter` is initially set to 0. Table is a global list that maintains associations between triples and already computed `UID`'s; it is also initially empty. The function `root(T)` extracts the label of the root of tree T.

We now give an example of the behavior of the procedure for an arithmetic expression.

Example 2.1: Consider the labeled tree necessary to store the arithmetic expression `(* (- (* x x) (* y y)) (+ (* x x) (* y y)))` which represents $(x^2 - y^2)(x^2 + y^2)$. The

“Table”, built by the `UID` procedure, contains

$$((x,0,0),1),\qquad ((y,0,0),3),\qquad ((-,2,4),5),\qquad ((\times,5,6),7),$$
$$((\times,1,1),2),\qquad ((\times,3,3),4),\qquad ((+,2,4),6),$$

and the tree in its full and compacted version is shown in Figure 2. ■

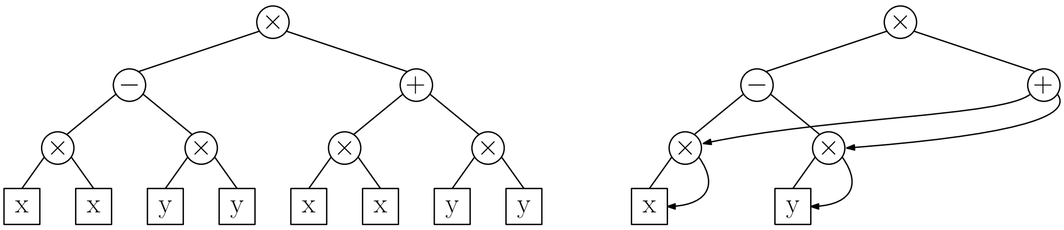

Figure 2: Tree and compacted tree associated with `(* (- (* x x) (* y y)) (+ (* x x) (* y y)))` computed by the `UID` procedure from Figure 1.

Motivated by this procedure, based on a post-order traversal of the tree, we define an ad hoc DAG-structure, which we call a *compacted binary tree*, that encodes the result of the compaction of the tree. The trees under consideration are *full binary* in the sense that their nodes have either 0 or 2 children. Furthermore, in the definition we refer to subtrees: a *fringe subtree* or short *subtree* is the tree which corresponds to a node and all its descendants. In this paper we only consider such subtrees.

Definition 2.2. *A* compacted binary tree *is a DAG computed by the* `UID` *procedure from a given full binary tree. Every edge leading to a subtree that has already been seen during the traversal is replaced by a new kind of edge, a* pointer*, to the already existing subtree. The* size *of the compacted binary tree is defined by the number of its internal nodes.*

In the sequel we will only consider full binary trees and their compacted forms. Thus, the term *compacted trees* means compacted binary trees. In Figure 3, we represent all compacted trees of size $0, 1$, and 2.

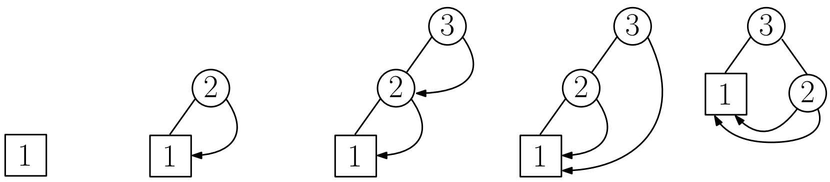

Figure 3: All compacted trees of size $n = 0, 1, 2$. The labels in the nodes are the uids of the corresponding subtrees. But note: The labels are not belonging to the combinatorial objects. Compacted trees are unlableled graphs.

The subclass of DAGs we are interested in is strongly influenced by properties of trees. In particular, compacted trees are connected and plane. The out-degree[2] of each node is equal

[2]For the terms out- and in-degree, source, sink, and so on, we interpret an undirected edge as directed away from the root, in accordance with a node-child relation.

to 2, except for the unique sink (leaf) for which it is 0. Furthermore, there is a unique source, which is the root.

The latter properties are induced by the full binary tree structure. Next, we treat the specific properties of the `UID` procedure. The result of the algorithm strongly depends on the chosen traversal. In this case the post-order traversal is used – but one could also consider a different one. There are two important observations. First of all, it has an important consequence on the pointers:

Proposition 2.3. *In a compacted tree the pointers only point to previously discovered trees.*

In other words, the ordering imposed by the traversal restricts the possible choices of the pointers.

Definition 2.4. *For any compacted tree of size n, the* spine *is the structure (with n nodes) obtained from the compacted tree by deleting all pointers and the leaf.*

In the Figure 4, from left to right, we see a compacted tree (without details on the pointers) and its spine. Furthermore, every distinct subtree is stored only once. In terms of the corresponding compacted trees this translates into uniqueness of every subtree.

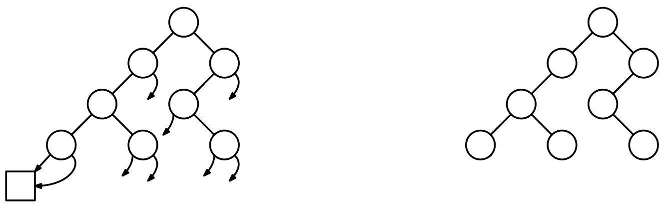

Figure 4: A compacted tree and its spine.

3 Main results

Before being able to state our main results we have to define further combinatorial classes. Indeed, the uniqueness condition for compacted trees caused some difficulties in their enumeration. So, we will first analyze a simpler class where we drop this condition.

Definition 3.1. *A* relaxed compacted binary tree *(short* relaxed binary tree*, or just* relaxed tree*), of size n is a directed acyclic graph consisting of a binary tree with n internal nodes, one leaf, and n pointers. It is constructed from a binary tree of size n, where the first leaf in a post-order traversal is kept and all other leaves are replaced by pointers. These links may point to any node that has already been visited by the post-order traversal.*

Obviously, the notion of spine adapts to the class of relaxed trees.

In fact, let us give another way to interpret compacted trees: compacted trees are relaxed trees with the restriction that all nodes in the spine are the roots of unique subtrees of the full tree. Note that this condition does not hold for all relaxed trees. In particular compare Figure 5 for the smallest relaxed tree which is not a compacted tree.

The asymptotic enumeration of relaxed trees is still too complicated. We will derive recurrence relations for their counting sequence as well as for the counting sequence of compacted trees. In order to obtain asymptotic results, we restrict the right height.

Figure 5: Left: the smallest relaxed tree that is not a compacted tree; right: the corresponding unique compacted tree.

Definition 3.2. *For any relaxed tree, we define its* right height *to be the maximal number of right edges on any path from the root to another node in the spine (of the relaxed or compacted tree under consideration). The* level *of a node is the number of right edges on the path from the root to this node.*

Figure 6 introduces an example and a natural way of representing a relaxed tree in order to emphasize these notions. It proves convenient to rotate the trees by 45 degrees.

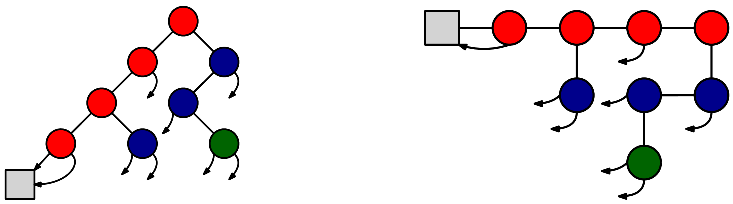

Figure 6: A compacted tree with right height 2. Nodes of level 0 are colored in red, nodes of level 1 in blue, and the node of level 2 in green.

Bounding the right height defines a sequence of classes which follows a recursive construction principle. We will eventually exploit this structure and obtain our main results, the asymptotic number of relaxed trees with n internal nodes and the analogous result for compacted trees.

Theorem 3.3 (Asymptotics of relaxed trees with bounded right height)**.** *The number $r_{k,n}$ of relaxed trees with right height at most k is for $n \to \infty$ asymptotically equivalent to*

$$r_{k,n} \sim \gamma_k n! \left(4\cos\left(\frac{\pi}{k+3}\right)^2\right)^n n^{-k/2},$$

where γ_k is a positive constant which is independent of n.

Theorem 3.4 (Asymptotics of compacted trees with bounded right height)**.** *The number $c_{k,n}$ of compacted trees with right height at most k is for $n \to \infty$ asymptotically equivalent to*

$$c_{k,n} \sim \kappa_k n! \left(4\cos\left(\frac{\pi}{k+3}\right)^2\right)^n n^{-\frac{k}{2}-\frac{1}{k+3}-\left(\frac{1}{4}-\frac{1}{k+3}\right)\cos\left(\frac{\pi}{k+3}\right)^{-2}},$$

where κ_k is a positive constant which is independent of n.

Therefore, we can also answer the question (at least asymptotically) of how many relaxed trees are actually compacted trees. Combining Theorems 3.3 and 3.4 we get the following result.

Corollary 3.5 (Proportion of compacted among relaxed trees)**.** *Let $c_{k,n}$ ($r_{k,n}$) be the number of compacted (relaxed) binary trees with right height at most k. Then, for $n \to \infty$ we have*

$$\frac{c_{k,n}}{r_{k,n}} \sim \frac{\kappa_k}{\gamma_k} n^{-\frac{1}{k+3}-\left(\frac{1}{4}-\frac{1}{k+3}\right)\frac{1}{\cos^2\left(\frac{\pi}{k+3}\right)}} = o\left(n^{-1/4}\right).$$

Thus, the number of compacted trees among relaxed trees for large n is negligible. This result quantifies the restriction of uniqueness of subtrees in compacted trees.

4 On the structure of compacted trees

In this section we will discuss some basic observations concerning the structure of compacted trees. First note that pointers may point to nodes lying outside the subtree of the pointer's start node (compare with Figures 2 and 3). Such subtrees of compacted trees cannot be compacted trees themselves. For this reason, we define the concept of *c-subtrees*.

Definition 4.1. *A* c-subtree *is the subgraph of a compacted tree induced by a node and all its descendants. A* cherry *is a c-subtree where both children of the root are pointers.*

A cherry is, in a sense, the "minimal" construction to create a new (unique) subtree. It consists of a node and two pointers, which point to already found *c*-subtrees during the traversal process. An example is given in Figure 3: In the rightmost tree, the *c*-subtree with the root node labeled by 2 is a cherry. Such a cherry is not a compacted tree in the sense of Definition 2.2, as the root node has two pointers which point to an external structure. It represents, however, a subtree and it corresponds in a unique way to the compacted tree of this subtree. The only compacted tree of size 1 is also given in the same figure.

With this terminology we are able to analyze some aspects of the DAG-structure of compacted trees. First, we look at the spine.

Lemma 4.2. *The spine of a compacted tree of size n is a binary tree of size n.*

Proof. Obviously, by deleting the leaf and the pointers we get a rooted, acyclic graph. It remains to show that this graph is connected. Assume that there exists a pointer which is the only connection between two parts of the compacted tree. By the `UID` procedure a pointer corresponds to a multiple occurrence of a subtree. Therefore we get a contradiction, as this subtree must already exist in the tree and is, therefore, connected with the root via internal edges. □

Let us remark that the tree structure of a spine is binary in the sense that its nodes are either of out-degree 2, 1 (with two possibilities, either with a left child or with a right child), or 0.

Proposition 4.3. *From any binary tree of size n, we can build a compacted tree of size n, with the following operations:*

1. *Add a leaf as left child of the leftmost node of the binary tree.*
2. *Add pointers to every node such that every node except the leaf has out-degree 2.*

3. *Let the pointers point to internal nodes which are in post-order traversal before the root node (under consideration) such that the corresponding subtree is unique (not already existing).*

Every compacted tree of size n can be constructed this way.

Proof. A simple way to build a compacted tree by using the spine is the following one. Add the leaf to the leftmost node of the binary tree. Then traverse the binary tree by using the post-order traversal. Each time one meets a node with out-degree less than 2 one adds 1 or 2 pointers such that the uniqueness condition is not violated (e.g. to the last node that has been visited). Thus, starting from a DAG generated from the above operations, decompacting and compacting it again with the `UID` procedure one arrives at the same structure.

The last statement is obvious, since every compacted tree can be reconstructed from its spine using only the operations listed above. By Lemma 4.2 the spine has the same size as the compacted tree. □

The advantage of the previous proposition is that it gives us an alternative construction of compacted trees bypassing the `UID` procedure. Starting from a binary tree, we can construct several compacted trees by enriching this binary tree. So the function mapping compacted trees to its spine is not one-to-one.

A key observations is that cherries are the fundamental structures that guarantee the uniqueness of c-subtrees. Indeed, if a cherry violates the condition implicit in the third operation listed in Proposition 4.3, the structure is not a compacted tree according to our definition, but only a relaxed tree.

A different explanation why cherries are the crucial objects for uniqueness comes from the property that the compaction procedure generates an increasing set of elements, i.e. already seen subtrees. Here we mean that the next element is constructed by a new internal node and previous, already built elements. In particular, the first element is always a leaf, the second one is always an internal node with two leaves as children (a “classical cherry”). Then, as a third element one has an element with a new internal node and a cherry as its left child, or as its right child, or on both sides. How will further elements be built such that the uniqueness property is maintained? Let us focus on the bad ways to do so, *i.e.*, we ask: What is forbidden? There are two cases according to the type of the current node (in the post-order traversal of the tree):

- The current node is a cherry: The only forbidden way to place the two pointers is choosing an already generated subtree and letting the two pointers of the cherry point to the children of the subtree. Note that the children of an already generated subtree must have been generated before. Thus, for any already generated subtree there is one forbidden configuration for the placement of the pointers.

- The current node is not a cherry: In this case at least one edge is not a pointer. But then it can easily be seen by induction (on the size of c-subtrees) that the subtree of the corresponding child is unique when assigning pointers during a post-order traversal: If the pointer is the left edge, the right subtree has not been processed yet when the decision for the pointer's target is about to come. Otherwise, the left subtree is unique and its building (placing of all its pointers) ends just before the pointer being the right edge is processed. Hence, there is no restriction on placing the pointer since the current node will always generate an new subtree due to the unique first appearance of the subtree of one of its children.

This idea will be picked up in the next section and used to derive a recurrence relation for the number of compacted trees of size n. Besides, it shows that we have to be careful only when dealing with nodes having two pointers (see Section 8).

5 Compacted trees of unbounded right height

Using the properties stated in the last section for compacted trees, we are now able to exhibit a combinatorial recurrence based on a decomposition of the structures under consideration.

5.1 A recurrence relation for compacted trees

Let c_n be the number of compacted binary trees of size n. Recall that Figure 3 showed all compacted trees of size $0, 1$ and 2. The first few terms of the sequence are given by

$$(c_n)_{n \geq 0} = (1, 1, 3, 15, 111, 1119, 14487, 230943, 4395855, 97608831, \ldots).$$

This sequence is found as sequence `A254789` in Sloane's Online Encyclopedia of Integer Sequences[3]. Let us mention that it appeared independently online during our work on this problem. In this section we solve the counting problem by deriving the first defining recurrence relation.

Suppose that we perform a post-order traversal on a tree and that already p c-subtrees have been discovered. Then the current node is the root of another c-subtree. Let $\Gamma_{n,p}$ denote the class of all c-subtrees of size n that may show up as such a c-subtree. Then we may think of the already compacted subtrees as an external pool of trees where our pointers can point to additionally when continuing our traversal. For an illustration see Figure 7. Note that the leaf is always part of this pool but not counted, and all subtrees in the pool must be constructed out of elements from the pool. In this sense the pool is closed in itself, and its evolution in the compaction procedure is an increasing sequence of sets.

Figure 7: The two cases of the pool construction of Theorem 5.1. The pool (circled elements) represents the already visited c-subtrees the pointers may point to. In the second case it may also point to the c-subtrees of the left sibling.

We define the size of the pool to be the number of distinct subtrees with at least one internal node. Thus, the pool for the trees in $\Gamma_{n,p}$ has size p and consists of $p + 1$ distinct c-subtrees. This artificially looking convention will simplify the later analysis.

[3] www.oeis.org

Theorem 5.1. *Let $n, p \in \mathbb{N}$, and $\Gamma_{n,p}$ as above. Moreover, we denote the cardinality of $\Gamma_{n,p}$ by $\gamma_{n,p}$. Then*

$$\gamma_{n+1,p} = \sum_{i=0}^{n} \gamma_{i,p}\, \gamma_{n-i,p+i}, \qquad \textit{for } n \geq 1, \tag{1}$$

$$\gamma_{0,p} = p + 1, \tag{2}$$

$$\gamma_{1,p} = p^2 + p + 1. \tag{3}$$

Proof. An element of $\Gamma_{n,p}$ consists of n internal nodes connected by $n-1$ internal edges. The remaining $n+1$ edges of the compacted binary tree are pointers (the possible edge to the leaf may be interpreted as a pointer). These must be chosen in such a way that no subtree is generated twice. Additionally, they may point either to a c-subtree of the pool or to a c-subtree of its left sibling, see Figure 7. The second condition is due to the post-order traversal of the tree by the `UID` procedure.

Now we can give a recursive decomposition of such trees. Let t be a c-subtree with $n+1$ nodes and a pool of size p. The root of t has a left and a right subtree attached to i and $n-i$, (for $i = 0, \dots, n$) internal nodes, respectively. Note that every internal node also represents a c-subtree. For the left child the pool remains the same as for its parent. However, for the right child the pointers may additionally point to c-subtrees of its left sibling. Hence, the pool is increased by the size of its left sibling. These considerations directly give Equation (1).

Next, let us consider the initial conditions (2) and (3). The c-subtrees with no internal nodes can be interpreted as pointers. These may point to any element of the pool, hence $\gamma_{0,p} = p + 1$.

The c-subtrees with 1 internal node are cherries whose both children are not internal nodes. Hence, they consist either of two pointers or of a leaf and a pointer. As the pool always contains a leaf, it is sufficient to consider the first case. Then these two pointers have $p+1$ possibilities each to point at. Among these $(p+1)^2$ cases are p which must be excluded as they are the ones already found in the pool. Note that these can be recreated by letting the pointers point to the same children as the ones found in the pool. Hence, we get

$$\gamma_{1,p} = (p+1)^2 - p = p^2 + p + 1. \qquad \square$$

Corollary 5.2. *The number c_n of compacted trees of size n is equal to $\gamma_{n,0}$.*

Obviously, by Theorem 5.1 the numbers $\gamma_{n,0}$ depend on the numbers $\gamma_{k,p}$ for all $0 \leq k \leq n$ and all $0 \leq p \leq n$. Thus their computation is cubic in time and quadratic memory.

Next, let us state a simplified problem, which also proves very difficult to solve, but is not as technical.

5.2 A recurrence relation for relaxed compacted trees

Let r_n be the number of relaxed trees of size n. The first few terms of the sequence are given by

$$(r_n)_{n \geq 0} = (1, 1, 3, 16, 127, 1363, 18628, 311250, 6173791, 142190703, \dots).$$

This sequence is given by the sequence A082161 in the OEIS. The latter counts the number of deterministic completely defined initially connected acyclic automata with 2 inputs and n transient unlabeled states and a unique absorbing state, see [20]. The bijection of these structures to our (enriched) trees is obvious, by traversing relaxed trees from the root to the leaf. We remark that the asymptotic behavior of the number of such structures seems not to be known.

Let $\delta_{n,p}$ be the number of relaxed c-subtrees of size n and a pool of size p. We directly get a recurrence relation for these numbers, that is directly linked to the one for $(\gamma_{n,p})_{n,p\in\mathbb{N}}$:

Corollary 5.3. *Let $n, p \in \mathbb{N}$, then*

$$\delta_{n+1,p} = \sum_{i=0}^{n} \delta_{i,p}\, \delta_{n-i,p+i}, \qquad \text{for } n \geq 1, \tag{4}$$

$$\delta_{0,p} = p + 1. \tag{5}$$

The number of relaxed trees of size n is equal to $\delta_{n,0}$.

Proof. This is a direct consequence of Theorem 5.1 and the fact that we dropped the uniqueness restriction enforced by (3). □

Note that the nature of the recurrence relation did not change compared to the one of the compacted case. Unfortunately, we were not able to find an explicit solution, or to continue from here. However, using our main results on trees of bounded right height we are able to determine the asymptotic growth of this sequence.

5.3 The asymptotic growth of unbounded compacted and relaxed trees

In order to better understand the asymptotic growth of compacted trees we first consider some simple bounds.

Lemma 5.4. *The number of compacted trees of size n satisfies the following bounds:*

$$n! \leq c_n \leq r_n \leq \frac{1}{n+1}\binom{2n}{n}\, n!.$$

Proof. Let us first consider the lower bound: Consider the subclass of chains. These are trees where the left child is always an internal edge and the right child is a pointer, see Figure 8. Let a_n be the number of chains with n internal nodes. The leaf is the only such object of size 0. Hence, we have $a_0 = 1$. A chain of size $n+1$ can be constructed from a chain of size n by appending a new root node with a pointer. The pointer has $n+1$ possible locations to point to. This implies, $a_{n+1} = (n+1)a_n$. We get the lower bound $a_n = n!$.

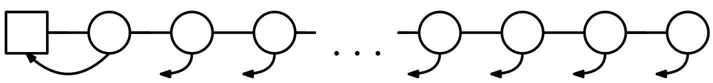

Figure 8: The number of compacted trees of size n of right height at most 0 is equal to $n!$.

Let us now focus briefly on the upper bound: Consider all possible spines. There are $\mathrm{Cat}_n = \frac{1}{n+1}\binom{2n}{n}$ (Catalan numbers) such structures, as they are binary trees. Next, note that a binary tree of size k has $k+1$ leaves. In our case these are pointers. By Proposition 2.3 pointers can only point to previously discovered trees. Hence, every pointer has at most k possibilities to point at. This proves the upper bound. □

The last result implies that the asymptotic growth of compacted trees satisfies $c_n = \mathcal{O}(n!\,4^n\,n^{-3/2})$, but it is also bounded from below by $n!$. This observation has two important implications. Firstly, an ordinary generating function for $(c_n)_{n\in\mathbb{N}}$ would have radius of convergence equal to zero. Hence, we will need to use exponential generating functions in order to ensure a non-zero radius of convergence. This idea will be used in the next sections. Secondly, combining it with our main result Theorem 3.4 we directly get:

Corollary 5.5. *The exponential growth of compacted and relaxed binary trees of size n is equal to 4^n, i.e.*

$$\lim_{n\to\infty} \frac{\log(c_n/n!)}{n} = 4, \qquad \textit{and} \qquad \lim_{n\to\infty} \frac{\log(r_n/n!)}{n} = 4.$$

Proof. Observe that for any k it holds that $c_{k,n} \leq c_n \leq r_n \leq n!\,\mathrm{Cat}_n$. Thus, the asymptotic form of the Catalan numbers and Theorem 3.4 show the claim. □

In the next section we return to compacted binary trees of bounded right height and start to capture their nature with exponential generating functions.

6 Operations on trees

We have seen in the previous sections that the numbers c_n and r_n are growing at least like $n!4^n$. Therefore we introduce exponential generating functions in order to get a non-zero radius of convergence. But then there arises a problem in the construction: exponential generating functions are designed for labeled objects, but we are dealing with unlabeled ones. Thus, we first investigate how the nature of exponential generating functions reflects the construction of such enriched trees.

The use of non-standard generating functions in the enumeration of DAGs is not new. Robinson [29] introduced the so-called "special generating function"

$$A(t) = \sum_{n\geq 0} a_n 2^{-\binom{n}{2}} \frac{t^n}{n!}$$

to derive nice expressions of such generating functions for labeled DAGs. This *ad hoc* generating function seems not applicable in our context, but exponential generating functions are.

For this purpose, we restrict ourselves to a subclass: relaxed trees of bounded right height, and we are going to derive their exponential generating functions. In this context we introduce the following notations: Let $\mathcal{R}$ be a combinatorial class. Its exponential generating function is given by $R(z) = \sum_{n\geq 0} r_n \frac{z^n}{n!}$ where r_n denotes the number of elements in $\mathcal{R}$ of size n.

Lemma 6.1. *(Adding a new root) Let $\mathcal{R}$ be a combinatorial subclass of relaxed trees, and let $\mathcal{S}$ be the combinatorial class whose elements consist of a new root node, with an element of $\mathcal{R}$ as its left child, and with a pointer as its right child. Then,*

$$S(z) = zR(z).$$

Proof. Consider a relaxed tree of $\mathcal{R}$ of size n. Adding a new root node with the considered tree as its left child creates a tree of size $n+1$. The new pointer has $n+1$ possibilities, in particular it may point to one of the n internal nodes or the leaf. On the level of generating functions this implies

$$S(z) = \sum_{n \geq 0} (n+1) r_n \frac{z^{n+1}}{(n+1)!} = zR(z). \qquad \square$$

With the help of this lemma, we are able to construct the generating function of relaxed trees of right height equal to 0. Let $\mathcal{R}_0$ be the respective combinatorial class, and $R_0(z) = \sum_{n\geq 0} r_{0,n} \frac{z^n}{n!}$ be the associated generating function.

Corollary 6.2. *The generating function of relaxed trees of right height equal to* 0 *is*

$$R_0(z) = \frac{1}{1-z}, \qquad \textit{and} \qquad r_{0,n} = n!.$$

Proof. Such a tree is either just a leaf of size 0 or it is constructed from an element of $\mathcal{R}_0$ by appending a new root node. Obviously, this construction does not increase the right height, and it constructs all such trees. On the level of generating functions this directly translates into

$$R_0(z) = 1 + zR_0(z).$$

Solving the equation and extracting coefficients gives the result. $\square$

This gives an alternative proof of the lower bound in Lemma 5.4. It nicely exemplifies how exponential generating functions model operations on compacted trees.

We proceed now with other operations on combinatorial classes and generating functions. The next two might seem "strange" at first glance, as they do not produce relaxed trees. However, they are the basic operations for the construction of other ones.

Lemma 6.3 (Adding/deleting the root while ignoring pointers)**.** *Let $\mathcal{R}$ be a class of relaxed trees. Let $\mathcal{I}$ be the class of objects obtained from $\mathcal{R}$ by adding a new root node* without pointer *(as its right child), and let $\mathcal{D}$ be the class obtained from $\mathcal{R}$ by deleting the root node but (if existent)* keeping its pointer*.*[4] *Then,*

$$I(z) = \int R(z)\, dz,$$
$$D(z) = \frac{d}{dz} R(z).$$

[4]This means in particular, that a single leaf, being root of a size 0 object, simply disappears. Furthermore, an object with a root having no pointers will become disconnected at the root. The pointers from the right to the left subtree remain. However, this construction will only be used when the root has a pointer.

Proof. Adding a new root node increases the size by one, whereas deleting it decreases it by one. Hence, elements of $\mathcal{R}$ of size n are in bijection with elements of $\mathcal{I}$ of size $n+1$ as well as with elements of $\mathcal{D}$ of size $n-1$, compare Figure 9. Therefore, we get

$$I(z) = \sum_{n\geq 0} r_n \frac{z^{n+1}}{(n+1)!} = \int R(z)\,dz,$$
$$D(z) = \sum_{n\geq 1} r_n \frac{z^{n-1}}{(n-1)!} = \frac{d}{dz}R(z).$$
□

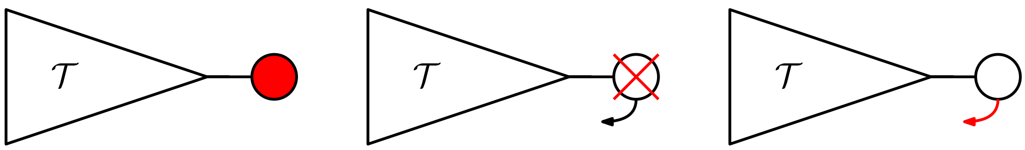

Figure 9: Adding a new root node without pointer, deleting a root node while preserving its (possible) pointer, and adding a new pointer to the existing root node.

These constructions can then be used to derive the following two operations:

Proposition 6.4 (Sequences and pointers)**.** *The generating function $S(z)$ corresponding to the class obtained by appending an arbitrary (possibly empty but finite) sequence of nodes to the root (each with one pointer) to a class $\mathcal{R}$ is given by*

$$S(z) = \frac{1}{1-z}R(z).$$

The generating function $P(z)$ of the class obtained by adding a new, additional pointer to the root nodes of the objects of a class $\mathcal{R}$ is given by

$$P(z) = z\frac{d}{dz}R(z) + r_0.$$

Proof. This is a direct consequence of the Lemmas 6.1 and 6.3, compare Figures 9 and 10. □

Remark 6.5: Note that when applying several consecutive sequence constructions as defined above, then the resulting structure looks like a single sequence construction. But we would get several factors $1/(1-z)$ in the generating function, though. This would only be correct if we set a marker after each application in order to remember where a sequence ends and the next one starts.

Alternatively, we may simply forbid consecutive sequence constructions. In particular, this means that $\mathcal{R}$ must be built in such a way that appending a sequence of nodes does not generate consecutive sequence constructions.

But all this is only a *caveat* in the usage of the sequence construction. When building relaxed and compacted trees, we never face consecutive sequence constructions, so there is no need to pay attention to it in our context. ■

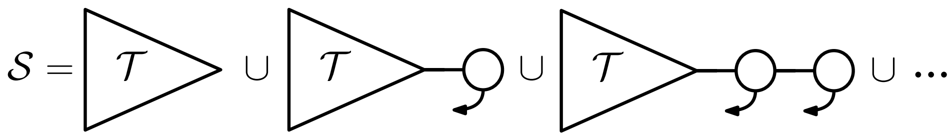

Figure 10: Appending a finite (possibly empty) sequence to the root node.

Now we have all operations needed to continue our investigation of trees with bounded right height. In the next sections we show how this calculus is used to derive differential equations for relaxed and compacted trees of bounded right height.

In the sequel, it will prove convenient to work with operators on generating functions. For this purpose, we will use the same letters for the operators as were used for the combinatorial classes (or generating functions).

7 Relaxed binary trees

We will now show how to use the calculus developed in Section 6 to derive ordinary differential equations for the exponential generating functions of relaxed trees of bounded right height. In this context we introduce the following notation: Let $\mathcal{R}$ be the combinatorial class of relaxed trees. Its exponential generating function is given by $R(z) = \sum_{n\geq 0} r_n \frac{z^n}{n!}$ where r_n denotes the number of elements in $\mathcal{R}$ of size n. We denote the class of relaxed trees of right height at most k by $\mathcal{R}_k$ and its corresponding exponential generating function by $R_k(z) = \sum_{n\geq 0} r_{k,n} \frac{z^n}{n!}$.

We have derived $R_0(z)$ in Corollary 6.2 as

$$R_0(z) = \frac{1}{1-z} = \sum_{n\geq 0} n! \frac{z^n}{n!}.$$

Let us now consider relaxed trees of right height *at most one*.

7.1 Relaxed trees of right height at most 1

Let $\mathcal{R}_1$ be the combinatorial class of relaxed trees with right height at most 1, compare Figure 11. The corresponding generating function is given by $R_1(z) = \sum_{n\geq 0} r_{1,n} \frac{z^n}{n!}$.

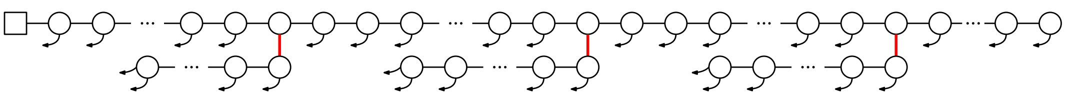

Figure 11: A relaxed tree from $\mathcal{R}_1$, i.e. with right height at most 1.

We will break the problem into smaller parts by decomposing $\mathcal{R}_1$ according to the following equation

$$R_1(z) = \sum_{\ell\geq 0} R_{1,\ell}(z), \qquad (6)$$

where $R_{1,\ell}(z)$ is the exponential generating function of relaxed binary trees with exactly ℓ right subtrees, i.e. ℓ right edges in the spine going from level 0 to level 1. Obviously, we have $R_{1,0}(z) = R_0(z) = \frac{1}{1-z}$. In order to get $R_{1,1}(z)$, we apply the previously developed constructions. An illustration of such a tree is shown in Figure 12.

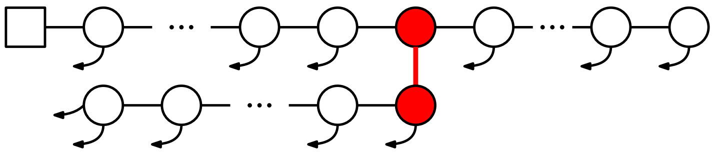

Figure 12: A relaxed tree with exactly one right edge in the spine.

Proposition 7.1. *The generating function of relaxed trees with exactly one right edge in the spine is given by*

$$R_{1,1}(z) = \frac{1}{1-z}\int \frac{1}{1-z} z\,(zR_{1,0}(z))'\,dz = \frac{z^2}{2(1-z)^3}.$$

Proof. The idea is to decompose the structure of $\mathcal{R}_{1,1}$ into smaller parts which are in bijection to constructible classes.

1. On level 0 there is a unique node with one right edge, see Figure 12. Before this node there is a possibly empty sequence of nodes corresponding to the sequence construction given by the operator $\mathcal{S}$. Call this the initial sequence. First consider a relaxed tree with empty initial sequence, see Figure 13.

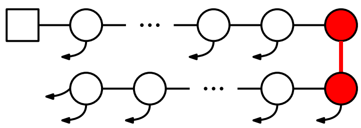

Figure 13: Step 1: An element of $\mathcal{R}_{1,1}$ with empty sequence of initial nodes on level 0.

2. On level 0, the left child of the unique node with two children (and without pointer) is followed by a sequence of nodes, whose pointers may only point to nodes of the sequence. This is an element of $\mathcal{R}_0$ and thus counted by $R_0(z)$.

   Furthermore, we see that the elements on level 1 form a sequence with a cherry as its last element. Its pointers may also point to nodes from the sequence discussed in the previous paragraph, which is in bijection with $\mathcal{R}_0$. By moving the $\mathcal{R}_0$-instance of level 0 to the end of the sequence on level 1 we get a sequence containing one special node which has two pointers. Then we delete the last node on level 0, compare with Figure 14.

   In terms of generating functions we get

$$\hat{R}_{1,0}(z) := \frac{1}{1-z}\underbrace{z\,(zR_{1,0}(z))'}_{\text{add root with two pointers}}. \tag{7}$$

   Note that due to the cherry every element has at least one internal node.

3. Furthermore, notice that the node on level 0 containing a right child (and not a right pointer) has no pointers. However, elements of the initial sequence may point to it. Therefore, we reinsert this node by adding it as a new root without pointer. The constructed object bijectively corresponds to the elements of $\mathcal{R}_{1,1}$ with empty initial sequence.

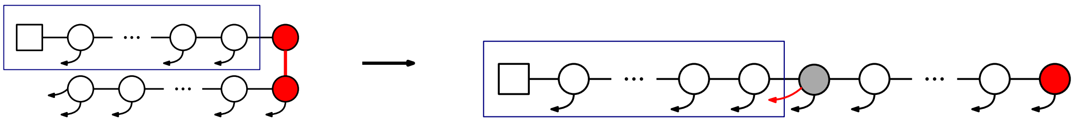

Figure 14: Step 2: Nodes of level 0 can only point to nodes on level 0 (left); moving these nodes to level 1 and deleting the remaining node at level 0 gives $\hat{R}_{1,0}(z)$ (right).

4. Finally, we append an initial sequence (cf. Step 1).

After those steps, the resulting object looks like shown in Figure 15: a sequence with two special nodes, one having no pointer, the other one having two pointers. The class of all such elements is in bijection with $\mathcal{R}_{1,1}$, as all the steps above can be reverted.

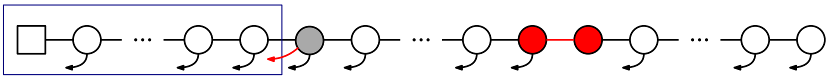

Figure 15: Step 4: The final sequence-like object bijectively corresponding to $\mathcal{R}_{1,1}$.

Now we have to translate the operations performed in the four steps into algebraic operations on generating functions. As already mentioned, after Step 2 the class of objects we get in that way has generating function $\frac{1}{1-z}z\,(zR_{1,0}(z))'$. The operation in Step 3 corresponds to integrating the generating function by Lemma 6.3. The final step is the application of the operator S of Proposition 6.4 and therefore generates a factor $\frac{1}{1-z}$, which completes the proof. ☐

The main idea of the previous proof was to cut and glue the $\mathcal{R}_{1,1}$-instance in such a way that a sequence-like object appears such that the process forms a bijection from $\mathcal{R}_{1,1}$ to the class of sequence-like objects of the form shown in Figure 15. This new object has the advantage of being constructible by the operations introduced in Section 6.

The previous decomposition captures all necessary mechanics to compute $R_{1,\ell}(z)$.

Corollary 7.2. *The generating function of relaxed trees $\mathcal{R}_{1,\ell}$ with exactly ℓ right edges in the spine from level* 0 *to level* 1 *is given by*

$$R_{1,\ell}(z) = \frac{1}{1-z}\int \frac{1}{1-z}z\,(zR_{1,\ell-1}(z))'\,dz, \qquad \ell \geq 1,$$
$$R_{1,0}(z) = R_0(z) = \frac{1}{1-z}.$$

Proof. By cutting at the first right edge from level 0 to level 1, we observe a decomposition into an initial sequence, a right edge from level 0 to level 1 with its two endnodes being a sequence on level 1 and an instance counted by $R_{1,\ell-1}(z)$. The decomposition is exhibited in Figure 16. Thus, we may reuse the construction from Proposition 7.1 by replacing the initial value $R_{1,0}(z)$ by $R_{1,\ell-1}(z)$. ☐

Finally, we are able to combine the previous results to derive the generating function of $\mathcal{R}_1$. We need the classical notation of *double factorials*:

$$n!! := \prod_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} (n-2k), \qquad \text{for } n \in \mathbb{N}.$$

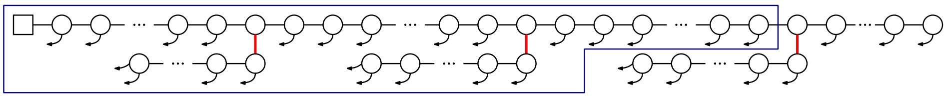

Figure 16: A recursive decomposition of elements from $\mathcal{R}_{1,\ell}$.

Theorem 7.3. *The exponential generating function of relaxed trees of right height at most* 1 *is D-finite and satisfies*

$$(1-2z)R_1'(z) - R_1(z) = 0, \qquad R_1(0) = 1.$$

The closed-form formula and the coefficients are given by

$$R_1(z) = \frac{1}{\sqrt{1-2z}}, \qquad r_{1,n} = (2n-1)!!.$$

Remark 7.4: For more on the general background of D-finite functions we refer to Stanley's excellent book [32]. Furthermore, note that the falling double factorials count many combinatorial families, see A001147. A bijective interpretation of this behavior was found by the last author in [36]. ■

Proof. We start with the result of Corollary 7.2. But instead of the integral representation, we use the following differential equation valid for $\ell \geq 1$:

$$(1-z)\left((1-z)R_{1,\ell}(z)\right)' = z\left(zR_{1,\ell-1}(z)\right)'.$$

Remembering the initial decomposition (6) and summing over all $\ell \geq 1$ we get

$$(1-z)\left((1-z)\left(R_1(z) - R_{1,0}(z)\right)\right)' = z\left(zR_1(z)\right)'.$$

Rearranging this equation and replacing $R_{1,0}(z)$ by $R_0(z)$ we get

$$(1-2z)R_1'(z) - R_1(z) - (1-z)\left((1-z)R_0(z)\right)' = 0. \tag{8}$$

Now, $R_0(z) = \frac{1}{1-z}$, hence the differential equation simplifies to

$$(1-2z)R_1'(z) - R_1(z) = 0.$$

Solving this equation by separation of variables yields the closed-form expression. Finally, the extraction of the coefficients is easy using $\frac{1}{\sqrt{1-4z}} = \sum_{n\geq 0}\binom{2n}{n}z^n$. □

7.2 Relaxed trees of right height at most 2

Let $\mathcal{R}_2$ be the combinatorial class of relaxed trees with right height at most 2, compare Figure 17. The corresponding generating function is given by $R_2(z) = \sum_{n\geq 0} r_{2,n}\frac{z^n}{n!}$.

In the same fashion as before, we will break the problem into smaller parts by decomposing $\mathcal{R}_2$ into

$$R_2(z) = \sum_{\ell \geq 0} R_{2,\ell}(z), \tag{9}$$

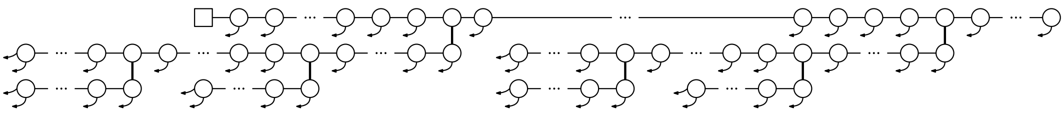

Figure 17: A relaxed tree from $\mathcal{R}_2$, i.e. with right height at most 2.

where $R_{2,\ell}(z)$ is the exponential generating function of relaxed trees of right height at most 2 with exactly ℓ right edges in the spine going from level 0 to level 1. Obviously, we have $R_{2,0}(z) = R_0(z) = \frac{1}{1-z}$.

Remark 7.5: Note that, as seen in the sequel, the functions $R_{k,0}(z) = \frac{1}{1-z}$ are in fact the perturbation of the recurrence of differential equations we are currently building. Moreover, they also uniquely determine the initial condition of this recurrence. Therefore, we will sloppily call these functions as well as others in the same role "initial conditions". This should not be confused with the initial conditions of the differential equations themselves. Those do not play any role in our arguments, so the risk of confusion should be low. ∎

Proposition 7.6. *The exponential generating function of relaxed trees of right height at most* 2 *with exactly one right edge from level* 0 *to level* 1 *in the spine satisfies*

$$(1-2z)\left((1-z)R_{2,1}(z)\right)'' - \left((1-z)R_{2,1}(z)\right)' - \left(z(zR_{2,0}(z))'\right)' = 0. \tag{10}$$

Proof. The main idea is to decompose an object of $\mathcal{R}_{2,1}$ again into 4 parts (compare with Figure 18): an initial sequence, the first right edge from level 0 to level 1, the sequence on level 0 after this right edge, and an instance of $\mathcal{R}_1$ starting on level 1 after this right edge. Then we use the same transformation idea as in the proof of Proposition 7.1. We take the sequence on level 0 after the right edge and move it to the end of the $\mathcal{R}_1$-instance. Note that this is legitimate concerning the pointers. But it generates a node with two pointers within an instance of $\mathcal{R}_1$. With respect to this $\mathcal{R}_1$-instance this change happens on its top level to the very left.

We can now delete the initial sequence and the level 0 node of the right edge, as they can be created again by known operations. Let $\mathcal{F}$ be the class of objects obtained after performing the above operations and $F(z)$ be its generating function. Schematically, this class is shown in the bottom of Figure 18. By Lemma 6.3 and Proposition 6.4 we get

$$F(z) = ((1-z)R_{2,1}(z))'.$$

Note that $F(z)$ is associated to structures with right height at most 1. It is nearly an instance of $\mathcal{R}_1$. There are only two differences:

First, it has a special construction after its last right edge. With respect to the differential equation (8), which corresponds to the class $\mathcal{R}_1$, this changes the initial condition $\mathcal{R}_0$ (recall Remark 7.5!). Due to linearity, we can reuse this specification by replacing the initial condition. On the level of generating functions this corresponds to replacing $R_0(z)$ by $\frac{1}{1-z}(zR_{2,0}(z))'$, because a (possibly empty) sequence is followed by a node with a double pointer and an instance of $R_{2,0}(z)$, which is in this case another sequence (compare with

Figure 14). Let $\mathcal{G}$ be the corresponding combinatorial class and $G(z)$ its generating function. By (8) we have

$$(1-2z)G'(z) - G(z) - (1-z)((1-z)G_0(z))' = 0, \qquad \text{with} \qquad G_0(z) = \frac{1}{1-z}(zR_{2,0}(z))'.$$

Second, due to the unique right edge from level 0 to level 1, every object in $\mathcal{F}$ has at least one particular node, namely the red node on level 1 (compare the transformation shown in Figure 18). Let us describe the unfavourable case we need to avoid, namely that there is no such node. Looking back at the beginning of the transformation, this case is equivalent to the fact that the subtree on levels 1 and 2 is empty, or in other words, the right edge going from level 0 to level 1 (red egde in Figure 18) is only a pointer. During the transformation process the $\mathcal{R}_0$-instance at the end of level 0 is moved to level one, where it forms then an $\mathcal{R}_0$-instance with an additional pointer, namely the above-mentioned pointer being the red edge in Figure 18 in the unfavourable case. The generating function of such structures is $(zR_{2,0}(z))'$, as we start with an $\mathcal{R}_0$-instance (which can equivalently be regarded as an $\mathcal{R}_{2,0}$-instance), add a new root and then delete this new root, but keep its pointer. Hence, in order to correct for the unfavourable case we need to subtract $(1-z)G_0(z)$. We get

$$F(z) = G(z) - (1-z)G_0(z).$$

This yields

$$G(z) = ((1-z)R_{2,1}(z))' + (zR_{2,0}(z))'. \tag{11}$$

Finally, putting everything together some straightforward calculations show (10). □

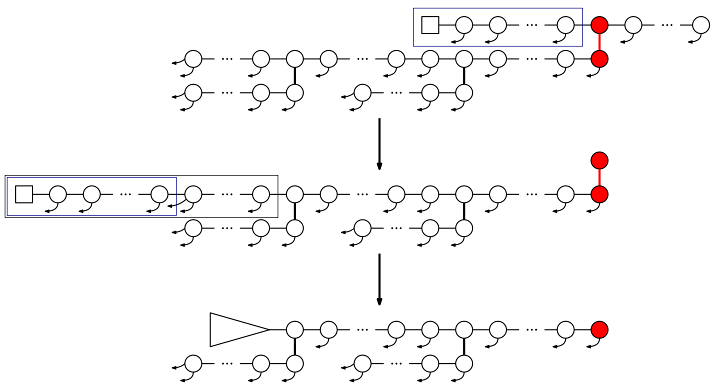

Figure 18: Transforming a structure of $\mathcal{R}_{2,1}$ into an instance of $\mathcal{R}_1$.

As in the $R_{1,\ell}(z)$ case, we get $R_{2,\ell}(z)$ for $\ell \geq 2$ by a recursive application of the previous arguments.

Corollary 7.7. *The generating function of relaxed trees with right height at most* 2*, and exactly* ℓ *right edges in the spine from level* 0 *to level* 1 *is given by*

$$(1-2z)\,((1-z)R_{2,\ell}(z))'' - ((1-z)R_{2,\ell}(z))' - (z(zR_{2,\ell-1}(z))')' = 0, \qquad \ell \geq 1,$$
$$R_{2,0}(z) = R_0(z) = \frac{1}{1-z}.$$

Proof. By cutting at the first right edge from level 0 to level 1, we observe a decomposition into an initial sequence, a right edge from level 0 to level 1 with 2 nodes, a sequence on level 1 and an instance counted by $R_{2,\ell-1}(z)$. Thus, we may reuse the construction from the proof of Proposition 7.6 by replacing the initial value $R_{2,0}(z)$ with $R_{2,\ell-1}(z)$. $\square$

Note that for the final result it is crucial that we found homogeneous differential equations.

Theorem 7.8. *The exponential generating function of relaxed trees of right height at most* 2 *is D-finite and satisfies*

$$(z^2 - 3z + 1)R_2''(z) + (2z - 3)R_2'(z) = 0, \qquad\qquad R_2(0) = 1, \; R_2'(0) = 1.$$

A closed-form formula and the coefficients are given by

$$R_2(z) = -\frac{2}{\sqrt{5}} \operatorname{artanh}\left(\frac{2z-3}{\sqrt{5}}\right) - \frac{1}{\sqrt{5}}\left(\log\left(\frac{7+3\sqrt{5}}{2}\right) - \pi i\right),$$
$$r_{2,n} = \frac{(n-1)!}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{2n} - \left(\frac{1-\sqrt{5}}{2}\right)^{2n}\right).$$

Proof. Again, let us take the result of Corollary 7.7 and sum over all $\ell \geq 1$, while remembering the decomposition (9). By linearity this gives

$$(1-2z)\left((1-z)(R_2(z) - R_{2,0}(z))\right)'' - \left((1-z)(R_2(z) - R_{2,0}(z))\right)' - \left(z(zR_2(z))'\right)' = 0. \quad (12)$$

A simplification gives

$$(z^2 - 3z + 1)R_2''(z) + (2z-3)R_2'(z) - (1-2z)((1-z)R_{2,0}(z))'' + ((1-z)R_{2,0}(z))' = 0.$$

Inserting the initial value $R_{2,0}(z) = \frac{1}{1-z}$ we get the D-finite expression. The correctness of the closed-form formula can then be easily checked with a computer algebra system.

In order to extract the coefficients of $R_2(z)$ we observe that the differential equation can be simplified further by an integration with respect to z. Thus, it is equivalent to

$$(z^2 - 3z + 1)R_2'(z) = 1, \qquad\qquad R_2(0) = 1,$$

as $R_2'(0) = 1$. Next, observe that as we are dealing with exponential generating functions, the derivative is just a shift on the level of coefficients. In other words, $[z^n]R_2(z) = [z^{n-1}]R_2'(z)$. Therefore, a partial fraction decomposition enables a direct extraction of the coefficients. $\square$

7.3 Relaxed trees of right height at most k

The approach from the previous section can be generalized to an arbitrary bound $k \geq 2$ for the right height. Let $R_k(z) = \sum_{n\geq 0} r_{k,n} \frac{z^n}{n!}$ be the corresponding generating function. The idea is to use the previous construction, and to derive a differential equation for $R_k(z)$ from the one of $R_{k-1}(z)$.

We introduce a family of linear differential operators L_k, $k \geq 1$, which describe the differential equations constructed for $R_k(z)$. Let D denote the differential operator $\frac{d}{dz}$ and 1 the identity operator, i.e. $1 \cdot F(z) = F(z)$. For example, $(D \cdot z)(F(z)) = \frac{d}{dz}(zF(z))$. We want to stress at this point that the operators are in general not commutative.

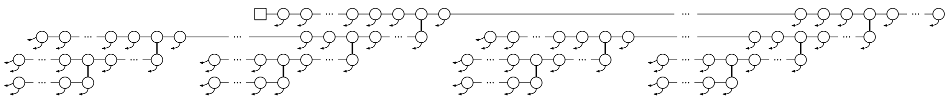

Figure 19: A relaxed tree from $\mathcal{R}_3$, i.e. with right height at most 3.

Theorem 7.9 (Differential operators)**.** *Let $(L_k)_{k\geq 0}$ be a family of differential operators given by*

$$\begin{aligned}
L_0 &= (1-z)\cdot 1,\\
L_1 &= (1-2z)D - 1,\\
L_k &= L_{k-1}\cdot D - L_{k-2}\cdot D^2 z, \qquad k\geq 2.
\end{aligned}$$

Then the exponential generating function $R_k(z)$ of relaxed binary trees with right height at most k satisfies for $k \geq 1$

$$L_k \cdot R_k(z) = 0. \tag{13}$$

Proof. For $k \geq 1$ we derive two families of operators: The differential operator L_k and an auxiliary operator H_k for the inhomogeneity such that

$$L_k \cdot R_k(z) = H_k \cdot R_0(z).$$

For $k = 1$ we derived in (8) the claimed form $L_1 \cdot R_1(z) = H_1 \cdot R_0(z)$ with $H_1 = L_0 D(1-z)$.

We continue with the case $k = 2$. The explicit differential operator is given in Theorem 7.8. We will now show how the operator can be constructed from the ones for $R_1(z)$ and $R_0(z)$ in the language of operators.

In Proposition 7.6 we have derived the necessary substitution to get the differential equation of $R_2(z)$ from the one of $R_1(z)$. The idea was to decompose $\mathcal{R}_2$ with respect to the number ℓ of right edges from level 0 to level 1, see Figure 18. This transformation creates an $\mathcal{R}_1$-like structure with a new initial condition $G_{0,\ell}(z)$ and the constraint not to be empty.

From (8) we get the generic differential equation for $\mathcal{R}_1$-like structures with generating function $G_\ell(z)$ as

$$L_1 \cdot G_\ell(z) = H_1 \cdot G_{0,\ell}(z).$$

First, the new initial condition is given by

$$G_{0,\ell}(z) = \left(\frac{1}{1-z} Dz\right) \cdot R_{2,\ell-1}(z).$$

Second, the $\mathcal{R}_1$-like class being in bijection to $\mathcal{R}_{2,\ell}$ cannot be empty, and the initial sequence on level 0 has to be appended. Thus, the substitution (11) has to be used where $R_{2,1}(z)$ is replaced by $R_{2,\ell}(z)$, and $R_{2,0}(z)$ by $R_{2,\ell-1}(z)$. This gives for $\ell \geq 1$

$$L_1\left((D(1-z))\cdot R_{2,\ell}(z) + (Dz)\cdot R_{2,\ell-1}(z)\right) - H_1\left(\left(\frac{1}{1-z}Dz\right)\cdot R_{2,\ell-1}(z)\right) = 0.$$

Summing over $\ell \geq 1$ and recalling that $R_2(z) = \sum_{\ell\geq 0} R_{2,\ell}(z)$ we get

$$L_1 \cdot D \cdot R_2(z) - L_0 \cdot D^2 \cdot z \cdot R_2(z) = L_1 \cdot D \cdot (1-z) \cdot R_{2,0}(z).$$

On the left we see the differential operator L_2 applied to $R_2(z)$ and on the right the inhomogeneity operator H_2 applied to $R_{2,0}(z)$. Inserting $R_{2,0}(z) = \frac{1}{1-z}$ shows the claim for $k = 2$.

Finally, for larger k, we can recycle the previous arguments for $k = 2$ and apply them recursively. This holds, as we may again cut an instance of $\mathcal{R}_k$ at the first right edge in the spine from level 0 to level 1 and decompose it in the repeatedly shown fashion, compare with Figure 18. Then the same reasoning as in Section 7.2 allows us to extract the differential equation of $R_k(z)$ from the one of $R_{k-1}(z)$ by

$$L_k = L_{k-1} \cdot D - H_{k-1} \cdot \frac{1}{1-z} \cdot D \cdot z, \qquad\qquad H_k = L_{k-1} \cdot D \cdot (1-z). \tag{14}$$

Hence, by induction the claim holds. □

Let us apply the last theorem and compute the first few differential equations.

$$(1-2z)\frac{d}{dz}R_1(z) - R_1(z) = 0,$$
$$(z^2-3z+1)\frac{d^2}{dz^2}R_2(z) + (2z-3)\frac{d}{dz}R_2(z) = 0,$$
$$(3z^2-4z+1)\frac{d^3}{dz^3}R_3(z) + (9z-6)\frac{d^2}{dz^2}R_3(z) + 2\frac{d}{dz}R_3(z) = 0,$$
$$-(z^3-6z^2+5z-1)\frac{d^4}{dz^4}R_4(z) - (6z^2-24z+10)\frac{d^3}{dz^3}R_4(z) - (6z-11)\frac{d^2}{dz^2}R_4(z) = 0.$$

The initial conditions of the differential equations can be obtained successively from lower order solutions. In particular, note that due to the construction the coefficients of $z^0, z^1, \ldots, z^{k+1}$ of $R_k(z)$ are the first $k+2$ elements of the counting sequence of relaxed trees, as a tree of size $k+1$ has always right height at most k. Thus with $R_k(z)$ we can enumerate all relaxed trees up to size $k+1$.

Next, we take a closer look at these operators.

Theorem 7.10 (Properties of L_k). *For any $k \in \mathbb{N}$, let L_k be as in Theorem 7.9. Let $\ell_{k,i}(z) \in \mathbb{C}[z]$ be such that*

$$L_k = \ell_{k,k}(z)D^k + \ell_{k,k-1}(z)D^{k-1} + \ldots + \ell_{k,0}(z). \tag{15}$$

Then we have

$$\begin{aligned}
&\ell_{k,0}(z) = 0,\\
&\ell_{k,1}(z) = \ell_{k-1,0}(z) - 2\ell_{k-2,0}(z),\\
&\ell_{k,i}(z) = \ell_{k-1,i-1}(z) - (i+1)\ell_{k-2,i-1}(z) - z\ell_{k-2,i-2}(z), \qquad 2 \leq i \leq k-1,\\
&\ell_{k,k}(z) = \ell_{k-1,k-1}(z) - z\ell_{k-2,k-2}(z).
\end{aligned}$$

The initial polynomials are $\ell_{0,0}(z) = 1-z$, $\ell_{1,0}(z) = -1$, and $\ell_{1,1}(z) = 1-2z$.

Proof. The initial polynomials are given by Theorem 7.9. The shape (15) of the operator L_k follows by induction using its recursive definition. Using an ansatz and comparing coefficients gives the recurrence relations for $\ell_{k,i}(z)$. $\square$

The asymptotic behavior (according to n) of the number $r_{k,n}$ of relaxed trees with right height at most k is governed by these differential equations. These differential equations belong to a known class [14, Chapter VII.9]. Consider an ordinary generating function of the kind

$$D^r Y(z) + a_1(z) D^{r-1} Y(z) + \cdots + a_r(z) Y(z) = 0, \tag{16}$$

where the $a_i \equiv a_i(z)$ are meromorphic in a simply connected domain Ω. Given a meromorphic function $f(z)$, let $\omega_\zeta(f)$ be the order of the pole of f at ζ, and $\omega_\zeta(f) = 0$ meaning that $f(z)$ is analytic at ζ.

Definition 7.11 (Regular singularity, [14, p. 519])**.** *The differential equation* (16) *is said to have a singularity at ζ if at least one of the $\omega_\zeta(a_i)$ is positive. The point ζ is said to be a* regular singularity *if*

$$\omega_\zeta(a_1) \le 1, \qquad \omega_\zeta(a_2) \le 2, \qquad \dots, \qquad \omega_\zeta(a_r) \le r,$$

and an irregular singularity *otherwise.*

Definition 7.12 (Indicial polynomial, [14, p. 520])**.** *Given an equation of the form* (16) *and a regular singular point ζ, the* indicial polynomial $I(\alpha)$ *at ζ is defined as*

$$I(\alpha) = \alpha^{\underline{r}} + \delta_1 \alpha^{\underline{r-1}} + \cdots + \delta_r, \qquad \text{where} \qquad \alpha^{\underline{\ell}} := \alpha(\alpha-1)\cdots(\alpha-\ell+1)$$

and $\delta_i := \lim_{z\to\zeta}(z-\zeta)^i a_i(z)$. The indicial equation at ζ *is the algebraic equation $I(\alpha) = 0$.*

The following technical lemma will be needed to derive the asymptotics for the solutions of the special type of differential equations given in Theorem 7.14.

Lemma 7.13. *Let $p_0, \dots, p_r \in \mathbb{C}[x]$ and consider the differential operator*

$$L = p_r D^r + \cdots + p_1 D + p_0.$$

Suppose that x is a simple factor of p_r, and suppose that for some $\alpha \in \mathbb{C}$, a solution of $Lf(x) = 0$ admits a generalized series solution $f(x) = \sum_{n\in\alpha+\mathbb{Z}} c_n x^n$. Then the coefficient sequence $(c_n)_{n\in\alpha+\mathbb{Z}}$ satisfies a recurrence of the form

$$\begin{array}{rlll} \Big(([x^1]p_r)(n-r+1) + ([x^0]p_{r-1})\Big) & n^{\underline{r-1}}\, c_n & & \\ + \quad [\cdots] & (n-1)^{\underline{r-2}}\, c_{n-1} & & \\ + \quad [\cdots] & (n-2)^{\underline{r-3}}\, c_{n-2} & & \\ + \quad \dots & & & \\ + \quad [\cdots] & c_{n-s} & = & 0, \end{array}$$

where $[\cdots]$ are certain polynomials in n and s is some fixed nonnegative integer.

Proof. We have $x^j D^i f = \sum_{n\in\alpha+\mathbb{Z}} c_n n^{\underline{i}} x^{n-i+j} = \sum_{n\in\alpha+\mathbb{Z}} c_{n+i-j}(n+i-j)^{\underline{i}} x^n$ for all $i,j\in\mathbb{N}$. Write $p_i = \sum_j p_{i,j}x^j$ for $i=0,\dots,r$, in the understanding that j runs through all integers, but $p_{i,j}$ is zero for all negative and almost all positive indices j. By assumption, we know that $p_{r,0}=0\neq p_{r,1}$.

It follows that $p_i D^i f = \sum_{n\in\alpha+\mathbb{Z}}\sum_j p_{i,j}c_{n+i-j}(n+i-j)^{\underline{i}}x^n$ for $i=0,\dots,r$. Then

$$Lf = \sum_{n\in\alpha+\mathbb{Z}}\sum_{i=0}^{r}\sum_j p_{i,j}c_{n+i-j}(n+i-j)^{\underline{i}}x^n = 0$$

implies, by comparing the coefficients of x^n,

$$0 = \sum_{i=0}^{r}\sum_j p_{i,j}c_{n+i-j}(n+i-j)^{\underline{i}} = \sum_j\sum_{i=0}^{r} p_{i,i+j}(n-j)^{\underline{i}}c_{n-j} \tag{17}$$

for all $n\in\alpha+\mathbb{Z}$.

Consider a fixed $j\in\mathbb{Z}$. From the definition $(n-j)^{\underline{i}} = (n-j)(n-j-1)\cdots(n-j-i+1)$ it follows that $(n-j)^{\underline{i}} \mid (n-j)^{\underline{i+1}}$ for every $i\in\mathbb{N}$. Therefore, if k is minimal such that $p_{k,k+j}\neq 0$, then $(n-j)^{\underline{k}} \mid \sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}}$.

Note also that for each fixed j, the polynomial $\sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}}$ is non-zero if and only if at least one of the coefficients $p_{i,i+j}$ are non-zero, because the falling factorials form a basis of the vector space of polynomials.

For $j<-r$, we have $i+j<0$ for all $i=0,\dots,r$, and therefore $p_{i,i+j}=0$ for all i and $\sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}} = 0$. Therefore there are no terms c_{n-j} with $j<-r$ present in equation (17).

For $j=-r$, we have $i+j<0$ for all $i=0,\dots,r-1$, and therefore $p_{i,i+j}=0$ for all these i. In addition, we have $p_{r,r-r}=p_{r,0}=0$ by assumption, so again $\sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}}=0$, and no term c_{n-j} with $j=-r$ is present in equation (17).

Next, for $j=-r+1$ we have $p_{r,r+(-r+1)}=p_{r,1}\neq 0$ by assumption, so the term $c_{n-(r-1)}$ *does* occur in equation (17). Moreover, since $p_{i,i+(-r+1)}=0$ for all $i<r-1$, we have $\sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}} = p_{r,1}(n-j)^{\underline{r}} + p_{r-1,0}(n-j)^{\underline{r-1}} = (p_{r,1}n+p_{r-1,0})(n+r-1)^{\underline{r-1}}$.

In general, for any $j>-r+1$, we have $p_{i,i+j}=0$ for all $i<-j$ and therefore $(n-j)^{\underline{-j}} \mid \sum_{i=0}^r p_{i,i+j}(n-j)^{\underline{i}}$. (The understanding here is that $(n-j)^{\underline{-j}}=1$ if $-j$ is not positive.) Substituting $n-r+1$ for n, we have shown the stated form of the recurrence. □

If ζ is a regular singularity of a differential equation, then all solutions of the differential equation behave for $z\to\zeta$ like $(z-\zeta)^\alpha\log(z-\zeta)^\beta$ for some $\alpha\in\mathbb{C},\beta\in\mathbb{N}$. The exponents α are roots of the indicial polynomial, and the exponents of the logarithmic terms are related to multiple roots of the indicial polynomial and roots at integer distances. More precisely, in our case the following theorem will be applicable. It is a variant of [14, Theorem VII.9] which works due to $\omega_\zeta(a_i)=1$ for all $i=1,\dots,r$.

Theorem 7.14. *Consider the differential equation (16) and a regular singular point ζ such that $\omega_\zeta(a_i)\le 1$ for all $i=1,\dots,r$, and $\delta_1 := \lim_{z\to\zeta}(z-\zeta)a_1(z)\ge 0$. Then, the vector space of all solutions defined in a slit neighborhood of ζ has a basis of r functions, where $r-1$ functions are of the form*

$$(z-\zeta)^m H_m(z-\zeta), \qquad m=0,1,\dots,r-2,$$

with functions H_m being analytic at 0 and satisfying $H_m(0) \neq 0$. The r-th basis function depends on δ_1:

1. *For $\delta_1 \in \{0, 1, \ldots, r-1\}$ it is of the form*
$$(z-\zeta)^{r-1-\delta_1} H(z-\zeta) \log(z-\zeta);$$
2. *For $\delta_1 \in \{r, r+1, \ldots\}$ it is of the form*
$$(z-\zeta)^{r-1-\delta_1} H(z-\zeta) + H_0(z-\zeta) \left(\log(z-\zeta)\right)^k, \qquad \text{with} \qquad k \in \{0,1\};$$
3. *For $\delta_1 \notin \mathbb{Z}$ it is of the form*
$$(z-\zeta)^{r-1-\delta_1} H(z-\zeta);$$

where H is analytic at 0, with $H(0) \neq 0$.

Proof. Due to $\omega_\zeta(a_i) \leq 1$ we get by the definition of the indicial polynomial that $\delta_i = 0$ for $i \geq 2$. Hence, it is given by
$$I(\alpha) = \alpha^{\underline{r}} + \delta_1 \alpha^{\underline{r-1}} = \alpha^{\underline{r-1}}(\alpha - r + 1 + \delta_1).$$
Therefore, the roots are $0, 1, \ldots, r-2$, and $r-1-\delta_1$.

Let us treat the consecutive range of roots $0, 1, \ldots, r-2$ first. Consider the equivalent recurrence relation for the coefficients $(c_n)_{n\in\mathbb{N}}$ of the series solution expanded at ζ. It has the form
$$I(n) y_n = \Phi(y_{n-1}, \ldots, y_{n-N}),$$
where $I(n)$ is the indicial polynomial, $N = \max_i(r - i + \deg(a_i(z)))$, and Φ is a linear operator with polynomial coefficients in n. Let α be a root of the indicial polynomial, and consider the sequence $(c_n)_{n\in\mathbb{N}}$ extended to $\mathbb{Z}$ with $c_n = 0$ for $n < \alpha$ for $\alpha = 0, 1, \ldots, r-2$. At $n = \alpha$ we have
$$0 \cdot y_\alpha = \Phi(0, \ldots, 0). \tag{18}$$
Hence, y_α can be chosen arbitrarily. By Lemma 7.13, for each choice the recurrence uniquely extends the sequence towards $+\infty$. Therefore, each root α gives rise to a different solution of our recurrence relation. The set of all these solutions is linearly independent. The consecutive range of zeros implies that the values $y_0, \ldots, y_{r-2}$ can be chosen arbitrarily, as they do not interfere with each other. Such a situation does not give rise to any logarithmic terms.

Next, let us treat the remaining basis solution associated to $r - 1 - \delta_1$.

In the first case, there is a multiple root of order 2. Then, the classical theory of linear differential equations implies the appearance of logarithmic terms, see [17, 18, 31, 37].

In the second case, the situation is analogous to (18): The solution starts to exist at $n = r - 1 - \delta_1$. But this solution then needs to be continued further, and at $n = 0$, we might have a problem. Then, there could emerge a logarithmic term or not. This depends on the specific problem. If the solution cannot be extended, then a logarithmic factor multiplied with the solution at $n = 0$ is added, see [18].

In the third case, the root does not interfere with the other solutions, as the difference with any other root is not an integer. Thus, it can be continued without problems, and has the claimed form. □

By Theorem 7.9 the differential equations associated to relaxed trees are of the kind (16). The roots of the leading term are under these conditions responsible for the singularities of the solutions. The dominant one is as usual the one closest to the origin. Our first aim is to show that for every bounded right height there exists a unique dominant singularity. For this purpose we start with the analysis of the polynomials $\ell_{k,i}(z)$. They are strongly connected to a famous family of polynomials: the *Chebyshev polynomials*, see, e.g., [12, Chapter 18] or [1, Chapter 22].

Definition 7.15 (Chebyshev polynomials). *The* Chebyshev polynomials of the first kind *$T_n(z)$ are defined by the recurrence relation*

$$\begin{aligned} T_0(z) &= 1, \\ T_1(z) &= z, \\ T_{n+2}(z) &= 2zT_{n+1}(z) - T_n(z). \end{aligned}$$

The Chebyshev polynomials of the second kind *$U_n(z)$ are defined by the recurrence relation*

$$\begin{aligned} U_0(z) &= 1, \\ U_1(z) &= 2z, \\ U_{n+2}(z) &= 2zU_{n+1}(z) - U_n(z). \end{aligned}$$

Lemma 7.16 (Transformed leading coefficient). *Let $\ell_{k,i}(z)$ be the coefficients of the operator L_k from Theorem 7.10. Then, for the leading coefficient we get*

$$\ell_{k,k}(z) = z^{\frac{k+2}{2}} U_{k+2}\left(\frac{1}{2\sqrt{z}}\right) = \sum_{n=0}^{\lfloor \frac{k+2}{2} \rfloor} (-1)^n \binom{k+2-n}{n} z^n,$$

where $U_k(z)$ are the Chebyshev polynomials of the second kind.

Proof. We start with the recurrence relation of $\ell_{k,k}(z)$ from Theorem 7.10. Replacing z by $\frac{1}{4z^2}$ and multiplying by $(2z)^{k+2}$, we get

$$(2z)^{k+2}\ell_{k,k}\left(\frac{1}{4z^2}\right) = 2z \cdot (2z)^{k+1}\ell_{k-1,k-1}\left(\frac{1}{4z^2}\right) - (2z)^k \ell_{k-2,k-2}\left(\frac{1}{4z^2}\right),$$

and we recognize the recurrence relation for the Chebyshev polynomials of the second kind for $(2z)^k\ell_{k-2,k-2}\left(\frac{1}{4z^2}\right)$, see [12, Section 18.9]. Transforming the initial conditions, gives $U_2(z)$ and $U_3(z)$ respectively.

The closed form is derived from the well-known formula $U_k(z) = \sum_{n=0}^{\lfloor \frac{k}{2} \rfloor} (-1)^n \binom{k-n}{n} (2z)^{k-2n}$. ∎

We will also need the following result on $\ell_{k,k-1}(z)$. Its structure is directly related to the one of $\ell_{k,k}(z)$.

Lemma 7.17 (Transformed $\ell_{k,k-1}(z)$). *For the coefficient $\ell_{k,k-1}(z)$ of the operator L_k from Theorem 7.10, we get*

$$\ell_{k,k-1}(z) = \frac{k}{2}\ell'_{k,k}(z),$$

for $k \geq 1$.

Proof. By Theorem 7.10 the claim holds for $k = 1$ and $k = 2$. We proceed by induction. Assume the claim holds for $1 \le i \le k$. Then, differentiating both sides of the defining equation of $\ell_{k,k}(z)$ given in Theorem 7.10 we get

$$\ell'_{k,k}(z) = \ell'_{k-1,k-1}(z) - z\ell'_{k-2,k-2}(z) - \ell_{k-2,k-2}(z).$$

Next, we apply the induction hypothesis and get

$$\ell'_{k,k}(z) = \frac{2}{k-1}\ell_{k-1,k-2}(z) - z\frac{2}{k-2}\ell_{k-2,k-3}(z) - \ell_{k-2,k-2}(z).$$

Finally, by rearranging the equation and utilizing the defining recurrence relation for $\ell_{k,k-1}(z)$ we prove (omitting the arguments)

$$\ell'_{k,k} = \frac{2}{k}\Big(\underbrace{\ell_{k-1,k-2} - z\ell_{k-2,k-3} - k\ell_{k-2,k-2}}_{=\ell_{k,k-1}}\Big) + \frac{1}{k}\Big(\underbrace{\ell_{k-1,k-1} - 2z\ell'_{k-2,k-2} + k\ell_{k-2,k-2}}_{=0}\Big),$$

where the last expression is equal to 0, as we know the polynomial $\ell_{k,k}(z)$ explicitly from Lemma 7.16. □

Chebyshev polynomials are well-studied objects. We summarize some important results (for our analysis) in the following lemma.

Lemma 7.18. *The roots of $\ell_{k,k}(z)$ are real, positive, and distinct. Let ρ_k be the smallest real root of $\ell_{k,k}(z)$. Then, $R_k(z)$ is singular at ρ_k and we have*

$$\rho_k = \frac{1}{4\cos^2\left(\frac{\pi}{k+3}\right)}.$$

Furthermore, ρ_k is not a root of $\ell_{k,k-1}(z)$.

Proof. The results follow from the well-known results on Chebyshev polynomials [12, Section 18.5]. In particular, the roots $x_{k,j}$ of $U_k(z)$ admit the closed-form expressions

$$x_{k,j} = \cos\left(\frac{j\pi}{k+1}\right).$$

This implies the closed-form expression of ρ_k. The last claim follows from the closed-form expression of $\ell_{k,k-1}(z)$ from Lemma 7.17 and the fact that the roots of Chebyshev polynomials are all simple.

Finally, note that $\rho_k \le \rho_{k-1}$ and $\rho_0 = 1$ is the singularity of $R_0(z) = \frac{1}{1-z}$. Let μ_k be the dominant singularity of $R_k(z)$. We prove by induction that $\mu_k = \rho_k$. Combinatorially, it is clear that $\mu_k \le \mu_{k-1} = \rho_{k-1}$. Furthermore μ_k must be related to $x_{k,j}$. Thus, as the the roots of the Chebyshev polynomials are interlacing we can only have $\mu_k = \rho_k$. □

Note that $\rho_0 = 1$, $\rho_1 = \frac{1}{2}$, and $\rho_2 = \frac{2}{3+\sqrt{5}}$ are exactly the singularities of $R_0(z)$, $R_1(z)$, and $R_2(z)$, respectively. Furthermore, with this information, we are finally able to characterize the indicial polynomials.

Proposition 7.19. *The indicial polynomial $I_k(\alpha)$ at ρ_k of the k-th differential equation is given by $I_k(\alpha) = \alpha^{\underline{k-1}}(\alpha - (\frac{k}{2} - 1))$.*

Proof. By Definition 7.12 we need to show that $\delta_i = 0$ for $i \geq 2$ and $\delta_1 = \frac{k}{2}$. The first claim holds by Lemma 7.18, as $\ell_{k,k-i}(z)/\ell_{k,k}(z)$ has no higher-order poles for all $i \geq 1$.

Let us reformulate the second claim:

$$\delta_1 = \lim_{z \to \rho_k} \frac{\ell_{k,k-1}(z)}{\frac{\ell_{k,k}(z)}{z-\rho_k}} = \frac{\ell_{k,k-1}(\rho_k)}{\ell'_{k,k}(\rho_k)} = \frac{k}{2}, \tag{19}$$

where the second equality sign holds because of de l'Hospital's rule and Lemma 7.18 (ρ_k is not a root of $\ell_{k,k-1}(z)$). The last equality holds by Lemma 7.17 $\square$

With the help of the following lemma, we are able to simplify the indicial polynomials further.

Lemma 7.20. *For $k \geq 2$ and $0 \leq i \leq \lfloor \frac{k-2}{2} \rfloor$ we have $\ell_{k,i}(z) \equiv 0$.*

Proof. Let us start with the cases $i = 0$ and $i = 1$. As defined in Theorem 7.10 we have $\ell_{k,0}(z) = 0$ for $k \geq 2$. The case $i = 1$ is valid for $k \geq 4$. Then, we have

$$\ell_{k,1}(z) = \ell_{k-1,0}(z) - \ell_{k-2,0}(z) = 0.$$

For the cases $i \geq 2$ we use induction on k. Assume the claim holds for $2, \dots, k-1$ and arbitrary i. Then, we have

$$\ell_{k,i}(z) = \ell_{k-1,i-1}(z) - (i+1)\ell_{k-2,i-1}(z) - z\ell_{k-2,i-2}(z) = 0.$$

In all three cases it is easy to check that $i \leq \lfloor \frac{k-2}{2} \rfloor$ and the induction hypothesis implies that these terms are equal to 0. $\square$

Hence, the differential equation of order k is actually a differential equation of order

$$\tilde{r} := \left\lceil \frac{k}{2} \right\rceil \qquad \text{for the function} \qquad \tilde{R}_k(z) := \frac{d^{\lfloor \frac{k}{2} \rfloor}}{dz^{\lfloor \frac{k}{2} \rfloor}} R_k(z).$$

In other words, we have

$$\ell_{k,k}(z) D^{\tilde{r}} \cdot \tilde{R}_k + \ell_{k,k-1}(z) D^{\tilde{r}-1} \cdot \tilde{R}_k + \cdots + \ell_{k,\lfloor \frac{k}{2} \rfloor}(z) \tilde{R}_k = 0. \tag{20}$$

Corollary 7.21. *Let $\tilde{I}_k(\alpha)$ be the indicial polynomial at ρ_k of the reduced differential equation (20). Then,*

$$\tilde{I}_k(\alpha) = \begin{cases} \alpha^{\underline{\tilde{r}-1}} (\alpha + 1), & \text{if } k \text{ even}, \\ \alpha^{\underline{\tilde{r}-1}} \left(\alpha + \frac{1}{2}\right), & \text{if } k \text{ odd}. \end{cases}$$

Proof. This is a direct consequence of Proposition 7.19. As only the order of the differential equation changed but not its coefficients, we get

$$\tilde{I}_k(\alpha) = \alpha^{\underline{\tilde{r}}} + \delta_1 \alpha^{\underline{\tilde{r}-1}} = \alpha^{\underline{\tilde{r}-1}} \left(\alpha - \left\lceil \frac{k}{2} \right\rceil + \frac{k}{2} + 1\right).$$

Considering the even and odd case separately yields the result. $\square$

After these technical steps, we can finally prove our first main result.

Proof of Theorem 3.3. By Lemma 7.18, ρ_k is the dominant singularity of $R_k(z)$. Furthermore, the pole of $\frac{\ell_{k,i}(z)}{\ell_{k,k}(z)}$ at $z = \rho_k$ for $i = 1, \ldots, k-1$ is of order one for $i \geq 1$. Thus, by Definition 7.11 it is a regular singularity of the differential equation.

Furthermore, by Corollary 7.21 the set of roots of the reduced indicial polynomial is $\{-1, 0, 1, \ldots, \tilde{r} - 2\}$ for even k and $\{-\frac{1}{2}, 0, 1, \ldots, \tilde{r} - 2\}$ for odd k. In both cases by Theorem 7.14, a basis in a slit neighborhood of ρ_k consists of the analytic functions

$$(1 - z/\rho_k)^s H_s(1 - z/\rho_k),$$

for $s = 0, \ldots, \tilde{r} - 2$ where H_s is analytic and nonzero at 0, and a singular function

$$\begin{cases} \frac{1}{1-z/\rho_k} H(1 - z/\rho_k) + G(1 - z/\rho_k) \log(1 - z/\rho_k), & \text{if } k \text{ is even,} \\ \frac{1}{\sqrt{1-z/\rho_k}} H(1 - z/\rho_k), & \text{if } k \text{ is odd,} \end{cases} \tag{21}$$

with functions G, H being analytic and nonzero at 0. These functions form a basis of the solution space of (20).

In order to obtain a basis of the solution space of the original differential equation (15), we need to integrate $\left\lfloor \frac{k}{2} \right\rfloor$ times. The analytic basis functions remain analytic and the singular one singular. As there is always just one singular function, and we know that $R_k(z)$ is singular at ρ_k, this function must be responsible for the asymptotic growth. We get a singular expansion for $z \to \rho_k$ of the kind

$$R_k(z) \sim \begin{cases} \tilde{\gamma}_k (1 - z/\rho_k)^{k/2-1} \log\left(\frac{1}{1-z/\rho_k}\right), & \text{if } k \text{ is even,} \\ \tilde{\gamma}_k (1 - z/\rho_k)^{k/2-1}, & \text{if } k \text{ is odd.} \end{cases}$$

Theorem 7.14 implies that $\tilde{\gamma}_k$ is a nonzero constant. As $R_k(z)$ is the generating function of a counting sequence, the sign of $\tilde{\gamma}_k$ must be such that the coefficients of the asymptotic main term of $R_k(z)$ are eventually positive. Finally, applying the transfer theorems [14], the claim holds with

$$\gamma_k = \frac{\tilde{\gamma}_k}{\Gamma(-k/2+1)} \text{ for odd } k \text{ and } \gamma_k = (-1)^{k/2+1} \Gamma\left(\frac{k}{2}\right) \tilde{\gamma}_k \text{ for even } k. \qquad \square$$

Let us comment on the even case. It is a priori not clear if this logarithmic term in (21) appears or not (if not we set $G \equiv 0$). But due to the appearance of the term with the polar singularity, the logarithmic term does not influence the asymptotic main term. Obviously, it plays a role for the error terms. For specific cases, we can of course answer this question. For $k = 2$, we have seen in Section 7.2 that there are no logarithmic terms. However, in this case, the reduced indicial polynomial is only of order 1, see Corollary 7.21. Therefore, the consecutive range of roots starting with 0 does not exist.

For $k = 4$, logarithmic terms appear. In this case we have the operator

$$(-z^3 + 6z^2 - 5z + 1)D^2 + (-6z^2 + 24z - 10)D + (11 - 6z)$$

and the expansion point ρ_4 that is a root of $-z^3 + 6z^2 - 5z + 1$. Then, the solution space is generated by the following two series:

$$
\begin{aligned}
&1\\
&-\tfrac{48\rho_4^2-267\rho_4+118}{14}(\rho_4-z)\\
&+\tfrac{249\rho_4^2-2340\rho_4+5560}{588}(\rho_4-z)^2\\
&-\tfrac{206442\rho_4^2-1141941\rho_4+502699}{7056}(\rho_4-z)^3\\
&+\mathcal{O}((\rho_4-z)^4),
\end{aligned}
$$

and

$$
\begin{aligned}
&(z-\rho_4)^{-1} && +0\,(z-\rho_4)^{-1}\log(z-\rho_4)\\
&+0 && -\tfrac{6\rho_4^2-33\rho_4+14}{7}\log(z-\rho_4)\\
&+\tfrac{159\rho_4^2-888\rho_4+440}{196}(\rho_4-z) && +\tfrac{3\rho_4^2-54\rho_4+194}{98}(\rho_4-z)\log(z-\rho_4)\\
&-\tfrac{2484\rho_4^2-14439\rho_4+9541}{1764}(\rho_4-z)^2 && -\tfrac{3834\rho_4^2-21183\rho_4+9221}{588}(\rho_4-z)^2\log(z-\rho_4)\\
&+\tfrac{4199487\rho_4^2-23629269\rho_4+12255158}{592704}(\rho_4-z)^3 && +\tfrac{3255\rho_4^2-175989\rho_4+805466}{49392}(\rho_4-z)^3\log(z-\rho_4)\\
&+\mathcal{O}((\rho_4-z)^4\log(z-\rho_4)).
\end{aligned}
$$

8 Compacted binary trees

After the successful application of exponential generating functions to relaxed trees of bounded right height, we will extend this method to compacted binary trees. In this section we solve the problem of finding the generating function of compacted trees of bounded right height. We denote the class of compacted trees of right height at most k by $\mathcal{C}_k$ and its corresponding exponential generating function by $C_k(z)=\sum_{n\geq 0} c_{k,n}\frac{z^n}{n!}$.

As every subtree in a relaxed tree of right height at most 0 is unique, by Corollary 6.2 we immediately get

$$C_0(z)=\frac{1}{1-z}.$$

8.1 The cherry operator

We start with the subclass $\mathcal{C}_1$ of compacted trees of right height at most 1. The same ideas as in Section 7.1 are used in the analysis. However, this case is more subtle as we have to guarantee uniqueness of the subtrees. The main observation in this context is that in order to establish uniqueness of the subtrees one needs to restrict the pointers of the cherries, see Proposition 4.3.

Consider a situation where the pointers of a cherry are pointing into a tree of size k. Thus, every pointer has $k+1$ possibilities ($+1$ due to the leaf). In a relaxed setting this would mean that there are $(k+1)^2$ different configurations.

In a compacted tree every internal node (or spine node) corresponds to a unique subtree. Therefore, the cherry has only $(k+1)^2-k$ different options, see also Theorem 5.1. Let us introduce the corresponding operator now.

Lemma 8.1 (Cherry operator)**.** *Let $\mathcal{C}$ be a class of compacted trees and $\mathcal{K}$ be the class obtained from $\mathcal{C}$ by adding a new node with two pointers, where the decompacted tree of this new node* *(left pointer is left child, right pointer is right child) is not part of $\mathcal{C}$. Then,*

$$\begin{aligned} K(z) &= z\,(zC(z))' - \int zC'(z)\,dz \\ &= z^2C'(z) + \int C(z)\,dz. \end{aligned}$$

Proof. The first term corresponds to the (unconstrained) operation of adding a root with two pointers, see (7). The second one is responsible for the correction, by deleting the number of subtrees which are already part of $\mathcal{C}$, see Figure 20:

Consider a tree of $\mathcal{C}$ of size k. The integrand creates a pointer attached to the root possibly pointing to all elements of the subtree. The integration operator adds a new root node without a pointer. By attaching the newly created pointer to this new root, and changing the pointer in the case of it pointing to the leaf by letting it point to the old root, we generate k new elements from this specific tree: A new root with a pointer to every internal node of the tree. This is exactly the number of elements which we need to subtract in order to ensure uniqueness.

The second expression results from an integration by parts of the first one. □

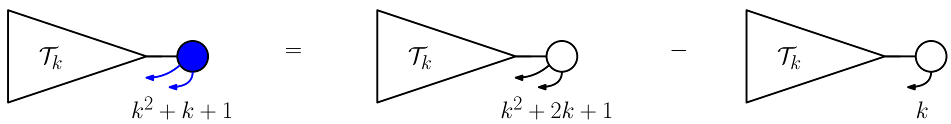

Figure 20: The construction of the cherry operator. The formulas below the pointers state the possible destinations of the pointers in the tree $\mathcal{T}_k$. The left tree is the desired one, the other ones are constructible ones.

Let us also define the corresponding operator which performs the previous operation:

$$K(C(z)) := z\,(zC(z))' - \int zC'(z)\,dz.$$

Next, we decompose $\mathcal{C}_1$ into

$$C_1(z) = \sum_{\ell \geq 0} C_{1,\ell}(z),$$

where $C_{1,\ell}(z)$ is the exponential generating function of compacted trees of right height at most 1 with exactly ℓ right edges in the spine going from level 0 to level 1.

Corollary 8.2. *The generating function of compacted trees with exactly ℓ right edges from level 0 to level 1 in the spine is given by*

$$\begin{aligned} C_{1,\ell}(z) &= \frac{1}{1-z}\int \frac{1}{1-z} K\left(C_{1,\ell-1}(z)\right)\,dz, \qquad \ell \geq 1, \\ C_{1,0}(z) &= \frac{1}{1-z}. \end{aligned}$$

Proof. The construction is analogous to the one of Corollary 7.2. The only difference is the use of the cherry operator in (7). □

Theorem 8.3. *The exponential generating function of compacted trees of right height at most* 1 *is D-finite and satisfies*

$$(1-2z)C_1''(z) - (3-z)C_1'(z) = 0, \qquad C_1(0) = 1, \quad C_1'(0) = 1.$$

The closed-form formula for $C_1'(z)$*, and the asymptotic behavior of the coefficients are given by*

$$C_1'(z) = \frac{e^{z/2}}{(1-2z)^{5/4}}, \qquad c_{1,n} = \frac{e^{1/4}}{\Gamma(1/4)} \frac{n!2^{n+1}}{n^{3/4}} \left(1 + \mathcal{O}\left(\frac{1}{n}\right)\right).$$

Proof. Summing the result of Corollary 8.2 for $l \geq 1$, interchanging summation, differentiation, and finally integration gives

$$(1-2z)C_1'(z) - C_1(z) - (1-z)\left((1-z)C_{1,0}\right)' + \int zC_1'(z)\,dz = 0.$$

Due to the remaining integral we differentiate both sides once more and get

$$(1-2z)C_1''(z) - (3-z)C_1'(z) - \Big((1-z)\left((1-z)C_{1,0}\right)'\Big)' = 0. \tag{22}$$

Inserting $C_{1,0}(z) = C_0(z) = \frac{1}{1-z}$ we get the claimed differential equation

$$(1-2z)C_1''(z) - (3-z)C_1'(z) = 0$$

It can be solved by separation of variables with respect to $C_1'(z)$. The asymptotic behavior of the coefficients follows then directly from this representation. □

8.2 Compacted trees of right height at most 2

We decompose $\mathcal{C}_2$ such that we get

$$C_2(z) = \sum_{\ell \geq 0} C_{2,\ell}(z),$$

where $C_{2,\ell}(z)$ is the exponential generating function of compacted trees of right height at most 2 with exactly ℓ right edges in the spine going from level 0 to level 1. Obviously, we have $C_{2,0}(z) = \frac{1}{1-z}$.

In the sequel we will use the operator calculus introduced in Section 6.

Proposition 8.4. *The generating function of compacted trees with right height at most* 2*, and exactly* ℓ *right edges from level* 0 *to level* 1 *in the spine is given by*

$$\begin{aligned} C_{2,\ell}(z) &= C_{2,\ell,A}(z) + C_{2,\ell,B}(z), \\ C_{2,\ell,A}(z) &= A(C_{2,\ell-1}(z)), \\ \bar{D}_2(C_{2,\ell,B}(z)) &= \bar{H}_2(C_{2,\ell-1}(z)), \end{aligned}$$

with the linear operators $A = S \cdot I \cdot S \cdot K$, $\bar{D}_2 = M_1 \cdot D \cdot S^{-1}$, $\bar{H}_2 = (H_1 - M_1) \cdot (D \cdot z + S \cdot K)$, *and* M_1 *and* H_1 *are given by*

$$M_1 = (1-2z)D^2 - (3-z)D, \tag{23}$$

$$H_1 = (1-z)^2 D^2 - 3(1-z)D + 1. \tag{24}$$

Proof. Using the same ideas as in the case of relaxed trees, we reduce the number of levels by deleting the initial sequence, and moving the last sequence to the end of the next lower level, see Figure 18. This produces a $\mathcal{C}_1$-like object with

- a new initial condition $\hat{C}_{2,0}(z)$ and
- the restriction of being non-empty.

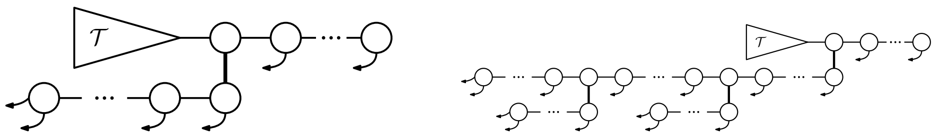

Figure 21: The 2 possible cases for $\mathcal{C}_2$ instances: Case (A) on the left, where level 2 does not exist; and case (B) on the right, where level 2 does exist.

In contrast to the relaxed case of $\mathcal{R}_{2,1}$ we need to distinguish whether level 2 exists or not (Figure 21). The different behaviors of single pointers and (double) cherry pointers are responsible for these two cases. Let $C_{2,1} \in \mathcal{C}_{2,1}$, $\hat{C}_{2,0}$ be the transformed object and $C_{2,0}$ be the part of level 0 located after the first right edge on level 0 ($\mathcal{T}$ in Figure 21).

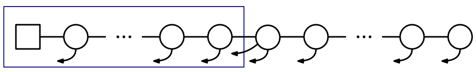

Figure 22: The new initial condition $\hat{C}_{2,0}(z)$. In case (A) the last sequence on level 1 cannot be empty, whereas in case (B) it may.

(A) Let $C_{2,1,A}(z)$ be the generating function of compacted trees belonging to Case (A) and $\mathcal{C}_{2,1}$. In this case level 2 does not exist (i.e. the tree also belongs to $\mathcal{C}_1$). Then we need to have a cherry on level 1, as this level is not allowed to be empty. This implies that the sequence of $\hat{C}_{2,0}$ shown in Figure 22 cannot be empty. Then, due to the previous reasoning on relaxed trees (cf. Proposition 7.1), and results on the trees in $\mathcal{C}_1$ (Corollary 8.2), we get the new initial condition for Case (A): Instead of $C_{1,0}(z)$ in (22) we must use

$$\hat{C}_{2,0,A}(z) := \frac{1}{1-z} K\left(C_{2,0}(z)\right) = \frac{z\left(zC_{2,0}(z)\right)' - \int zC'_{2,0}(z)\,dz}{1-z}.$$

This implies

$$C_{2,1,A}(z) = A(C_{2,0,A}(z)) := \frac{1}{1-z}\int \hat{C}_{2,0,A}(z)\,dz = \frac{1}{1-z}\int \frac{1}{1-z} K\left(C_{2,0,A}(z)\right)\,dz.$$

The first factor $\frac{1}{1-z}$ corresponds to the initial sequence on level 0, and the integral generates the level 0 node of the distinguished right edge. In anticipation of the subsequent result, we introduced the operator A.

(B) Let $C_{2,1,B}(z)$ be the generating function of this case. In this case level 2 exists. Then the last sequence of level 1, and therefore the initial sequence of $\hat{C}_{2,0}$, is allowed to be empty, see Figure 22. This means that no cherry was lost during the transformation into an instance of $\mathcal{C}_1$ as there is just one pointer pointing into $C_{2,0}$. Such a case is modeled by $(zC_{2,0}(z))'$. Combining it with the case of a non-empty sequence, we get the new initial condition of the case (B):

$$\hat{C}_{2,0,B}(z) := (zC_{2,0}(z))' + \frac{1}{1-z}K\left(C_{2,0}(z)\right) = \frac{(zC_{2,0}(z))' - \int zC_{2,0}'(z)\,dz}{1-z}.$$

The only difference to case (A) is the lack of the factor z in front of $(zC_{2,0}(z))'$.

By assumption we have nodes on level 2. This means that after the transformation into an instance of $\mathcal{C}_1$ we have nodes on level 1. Let $\bar{C}_1(z)$ be the exponential generating function of compacted trees of right height at most 1 with at least one node on level 1. Then

$$\bar{C}_1(z) = \sum_{\ell \geq 1} C_{1,\ell}(z) = C_1(z) - C_{1,0}(z).$$

The new operators, defined in (23) and (24), fulfill the same tasks as the ones from Theorem 7.9 for relaxed trees. From (22) we infer that $C_1(z)$ satisfies $M_1(C_1(z)) = H_1(C_0(z))$. Thus, for $\bar{C}_1(z)$ we get the following differential equation:

$$M_1(\bar{C}_1(z)) = M_1(C_1(z) - C_{1,0}(z)) = H_1(C_{1,0}(z)) - M_1(C_{1,0}(z)).$$

Then, the differential equation for $C_{2,1,B}(z)$ is given by

$$\begin{aligned}\bar{D}_2(C_{2,1,B}(z)) &:= M_1\left(((1-z)C_{2,1,B}(z))'\right) \\ &= H_1(\hat{C}_{2,0,B}(z)) - M_1(\hat{C}_{2,0,B}(z)) =: \bar{H}_2(C_{2,0}(z)),\end{aligned}$$

because we are able to reuse (22), with the new initial condition $\hat{C}_{2,0,B}(z)$ instead of $C_{1,0}(z)$. Its solution is equal to $((1-z)C_{2,1,B}(z))'$; see the process of Proposition 7.6. The new differential operator is thus given by

$$\bar{D}_2(F) = (2z^2 - 3z + 1)F''' - (z^2 - 10z + 6)F'' - (2z - 6)F'.$$

This process can now be continued recursively, like in Corollary 7.7. In order to derive $C_{2,2}(z)$, we replace $C_{2,0}(z)$ by $C_{2,1}(z)$, and so on. □

Using the last result we are able to characterize compacted trees of right height at most 2.

Theorem 8.5. *The exponential generating function of compacted trees of right height at most 2 is D-finite and satisfies*

$$\begin{aligned}(z^2 - 3z + 1)C_2'''(z) - (z^2 - 6z + 6)C_2''(z) - (2z - 3)C_2'(z) = 0,& \\ C_2(0) = 1,\ C_2'(0) = 1,\ C_2''(0) = 3.&\end{aligned}$$

Proof. The generating function $C_2(z)$ is decomposed into three parts:

$$C_2(z) = C_{2,0}(z) + C_{2,A}(z) + C_{2,B}(z),$$

where $C_{2,A}(z) = \sum_{\ell \geq 0} C_{2,\ell,A}(z)$, $C_{2,B}(z) = \sum_{\ell \geq 0} C_{2,\ell,B}(z)$, and the initial values $C_{2,0,A}(z) = C_{2,0,B}(z) = 0$. Summing the results of Proposition 8.4 for $\ell \geq 1$ gives

$$\begin{aligned} C_{2,A}(z) &= A(C_2(z)), \\ \bar{D}_2(C_{2,B}(z)) &= \bar{H}_2(C_2(z)). \end{aligned}$$

Finally, we get

$$\begin{aligned} \bar{D}_2(C_2) &= \bar{D}_2(C_{2,0} + C_{2,A} + C_{2,B}) \\ &= \bar{D}_2(C_{2,0}) + \bar{D}_2(A(C_2)) + \bar{H}_2(C_2), \end{aligned}$$

which gives the new differential operator M_2 and the inhomogeneity operator H_2:

$$M_2(C_2) := \bar{D}_2(C_2) - \bar{D}_2(A(C_2)) - \bar{H}_2(C_2) = \bar{D}_2(C_{2,0}) =: H_2(C_{2,0}). \tag{25}$$

Note that in analogy to (14) from the relaxed case we have here

$$H_2 = M_1 \cdot D \cdot (1 - z).$$

The computation of M_2 is direct with a computer algebra system (like Maple or Sage). $\square$

8.3 Compacted trees of right height at most k

Analogous to the case of relaxed trees in Section 7.3 the approach from the previous section can be generalized to an arbitrary bound $k \geq 2$ for the right height.

We introduce linear differential operators M_k, $k \geq 0$ which describe all differential equations constructed for $C_k(z)$. We use the same notation as in Section 7.3.

Theorem 8.6 (Differential operators)**.** *Let $(M_k)_{k \geq 0}$ be the family of differential operators given by*

$$\begin{aligned} M_0 &= (1 - z)D - 1, \\ M_1 &= (1 - 2z)D^2 - (3 - z)D, \\ M_k &= M_{k-1} \cdot D - M_{k-2} \cdot \left(D^2 \cdot z - zD\right), \qquad k \geq 2. \end{aligned}$$

Then the exponential generating function $C_k(z)$ of compacted binary trees with right height at most k satisfies

$$M_k \cdot C_k = 0.$$

Proof. The ideas are the same as the ones introduced in the proof of Theorem 7.9: In an instance of $\mathcal{C}_k$ we cut at the first right edge in the spine from level 0 to level 1. Then, the same decomposition as in the case $k = 2$ can be applied (like in Figure 18).

In particular, first generalizing the results of Propositition 8.4 we get

$$\begin{aligned}\bar{D}_k &= M_{k-1} \cdot D \cdot S^{-1}\\ \bar{H}_k &= \left(\bar{D}_{k-1} - M_{k-1}\right)(D \cdot z + S \cdot K),\end{aligned}$$

which together with (25) implies on the level of operators

$$\begin{aligned}M_k &= \bar{D}_k - \bar{D}_k \cdot A - \bar{H}_k\\ &= M_{k-1} \cdot D - M_{k-2} \cdot \left(D^2 z - zD\right).\end{aligned}$$

Recall for the second equality that D and I are inverse to each other, and that $S^{-1} = (1-z)$, and $K = zDz - IzD$. ☐

The first few differential equations are

$$(1-2z)\frac{d^2}{dz^2}C_1(z) + (z-3)\frac{d}{dz}C_1(z) = 0,$$

$$(z^2-3z+1)\frac{d^3}{dz^3}C_2(z) - (z^2-6z+6)\frac{d^2}{dz^2}C_2(z) - (2z-3)\frac{d}{dz}C_2(z) = 0,$$

$$(3z^2-4z+1)\frac{d^4}{dz^4}C_3(z) - (4z^2-18z+10)\frac{d^3}{dz^3}C_3(z) + (z^2-12z+14)\frac{d^2}{dz^2}C_3(z) + (z-3)\frac{d}{dz}C_3(z) = 0.$$

Theorem 8.7 (Properties of M_k)**.** *The operator M_k is a linear differential operator of order $k+1$ satisfying*

$$M_k = m_{k,k}(z)D^{k+1} + m_{k,k-1}(z)D^k + \ldots + m_{k,0}(z)D + m_{k,-1}(z),$$

where the $m_{k,i}(z)$ are polynomials given by the following recurrence relation for $k \geq 2$

$$\begin{aligned}m_{k,-1}(z) &= 0,\\ m_{k,0}(z) &= \begin{cases}-2z+3, & \text{for } k \text{ even},\\ z-3, & \text{for } k \text{ odd},\end{cases}\\ m_{k,i}(z) &= m_{k-1,i-1}(z) + (i+1)m_{k-2,i}(z)\\ &\quad + (z-i-2)m_{k-2,i-1}(z) - zm_{k-2,i-2}(z), \qquad 1 \leq i \leq k-1,\\ m_{k,k}(z) &= m_{k-1,k-1}(z) - zm_{k-2,k-2}(z),\\ m_{k,i}(z) &= 0, \qquad i > k.\end{aligned}$$

The initial polynomials are $m_{0,-1}(z) = -1$, $m_{0,0} = 1-z$, $m_{1,-1} = 0$, $m_{1,0} = z-3$, and $m_{1,1}(z) = 1-2z$. The leading coefficients $m_{k,k}(z)$ are the same as $\ell_{k,k}(z)$ from the relaxed case.

Proof. The proof is analogous to the one of Theorem 7.10. We omit the tedious calculations. ☐

It may seem artificial to start the second index at -1. However, the corresponding polynomials are equal to 0 except when $k = 0$. Thus, we are actually dealing with a differential equation of order k in $F'(z)$. Another advantage is that the leading polynomial $m_{k,k}(z)$, which is the same as the one in the relaxed case $\ell_{k,k}(z)$, has the same indices.

Following the approach used for relaxed trees, we then need to reveal the structure of the indicial polynomial.

In order to compute the value $\delta_1 = \lim_{z\to\zeta}(z-\zeta)a_1(z)$ (compare with (16)), we need the following result on $m_{k,k-1}(z)$.

Lemma 8.8 (Transformed $m_{k,k-1}(z)$)**.** *For the coefficient $m_{k,k-1}(z)$ of the operator M_k from Theorem 8.7, we get*

$$m_{k,k-1}(z) = z^{\frac{k+2}{2}} h_{k+2}\left(\frac{1}{2\sqrt{z}}\right),$$

where

$$h_k(z) = \frac{\left(k-3-2(k^2+k-2)z^2\right) T_k(z) + \left(1+2(k-1)z^2\right) U_k(z)}{2(z^2-1)},$$

and $T_k(z)$ and $U_k(z)$ are the Chebyshev polynomials of first and second kind, respectively.

Proof. From Theorem 8.7 we get the recurrence relation of $m_{k,k-1}(z)$:

$$m_{k,k-1}(z) = m_{k-1,k-2}(z) - z m_{k-2,k-3}(z) + (z-k-1) m_{k-2,k-2}(z).$$

Its structure is similar to the one of $m_{k,k}(z)$, but with an additional perturbation $(z-k-1)m_{k-2,k-2}(z)$. Transforming it in the same way as the one of $m_{k,k}(z)$ we get with

$$h_{k+2}(z) := (2z)^{k+2} m_{k,k-1}\left(\frac{1}{4z^2}\right),$$

for $k \geq 0$ the recurrence

$$h_{k+2}(z) = 2z h_{k+1}(z) - h_k(z) + \left(1 - 4z^2(k+1)\right) U_k(z).$$

From the theory of recurrences with constant coefficients (with respect to k) [19, Chapter 4] we get that the solution space is generated by $U_k(z), T_k(z), kU_k(z), kT_k(z), k^2U_k(z), k^2T_k(z)$. Making an ansatz and comparing coefficients gives the result. $\square$

Proposition 8.9. *Let $I_k(\alpha) = \alpha^{\underline{k+1}} + \delta_1\alpha^{\underline{k}} + \cdots + \delta_{k+1}$ be the indicial polynomial of the k-th differential equation, and let ρ_k be the smallest real root of $m_{k,k}(z)$. Then, we have $\delta_i = 0$ for $i > 1$, and $\delta_1 = \frac{m_{k,k-1}(\rho_k)}{m'_{k,k}(\rho_k)}$. Furthermore, we have*

$$\delta_1 = \frac{k}{2} + 1 - \frac{1}{k+3} - \left(\frac{1}{4} - \frac{1}{k+3}\right)\frac{1}{\cos^2\left(\frac{\pi}{k+3}\right)}.$$

The indicial polynomial is given by $I_k(\alpha) = \alpha^{\underline{k}}(\alpha - (k - \delta_1))$.

Proof. The first results are analogous to the ones in Proposition 7.19: First, because of Lemma 7.18 we have $\delta_i = 0$ for $i > 0$. Second, the expression for δ_1 is the same as (19), and follows from de l'Hospital's rule. Thus, the indicial polynomial is given by $I_k(\alpha) = \alpha^{\underline{k+1}} + \delta_1 \alpha^{\underline{k}}$.

We start with two simplifications for the root $x_k = \cos(\frac{\pi}{k+1})$ of $U_k(z)$ when inserted into $T_k(z)$. By the explicit expression $T_k(x) = \cos\left(k \arccos(x)\right)$, for $|x| \leq 1$, we get

$$T_k(x_k) = -\cos\left(\frac{\pi}{k+1}\right) = -x_k, \qquad \text{and} \qquad T_{k+1}(x_k) = -1.$$

First, we consider $m_{k,k-1}(z)$. By Lemma 8.8 we directly get

$$m_{k,k-1}(\rho_k) = -\rho_k^{\frac{k+2}{2}} \frac{(k-1)x_{k+2} - 2((k+2)^2 + k)x_{k+2}^3}{2(x_{k+2}^2 - 1)},$$

where $\rho_k = \frac{1}{4x_{k+2}^2}$, and recall that $U_k(x_k) = 0$.

Second, we consider the derivative of $m_{k,k}(z)$. Therefore, we use the following connection between Chebyshev polynomials of the first and second kind, see [12, Section 18.9]:

$$U_k'(z) = \frac{(k+1)T_{k+1}(z) - zU_k(z)}{z^2 - 1}.$$

Thus, by Lemma 7.16 we get

$$m_{k,k}'(\rho_k) = \frac{\rho_k^{\frac{k-1}{2}}}{4} \frac{k+3}{x_{k+2}^2 - 1}.$$

Combining these results shows the claim. □

We arrive at our main result for compacted binary trees, Theorem 3.4.

Proof of Theorem 3.4. The proof follows the same lines as the one of Theorem 3.3. In particular, the third case of Theorem 7.14 gives the asymptotic result, as δ_1 is irrational for all $k \in \mathbb{N}$. □

In contrast to relaxed trees, the asymptotics of compacted trees involves in general an irrational critical exponent. In Table 1 we list their first explicit values.

9 Conclusion

In this paper we solved the asymptotic counting problem for compacted and relaxed binary trees of bounded right height. In a compacted binary tree repeatedly occurring subtrees have been deleted and replaced by pointers to the first appearance, and hence every subtree is unique. By doing so, the tree structure is destroyed and replaced by a directed acyclic graph. In a relaxed binary tree the uniqueness condition of subtrees is omitted.

The difficulty of this counting problem is founded in the compaction procedure. A compacted binary tree of size n, where the size is the number of internal nodes, arises from a binary tree whose size is between n and 2^n. Our main results with regard to the general counting problem, are recurrence relations for compacted and relaxed binary trees in Theorem 5.1 and Corollary 5.3, respectively.

k	r_k	$r_k \approx$	$\kappa_k \approx$	α_k	$\alpha_k \approx$	$\gamma_k \approx$	β_k	$\beta_k \approx$
1	2	2.000	0.708	$-\frac{3}{4}$	-0.750	0.564	$-\frac{1}{2}$	-0.5
2	$4\cos(\frac{\pi}{5})^2$	2.618	0.561	$-\frac{6}{5} - \frac{1}{20\cos(\frac{\pi}{5})^2}$	-1.276	0.447	-1	-1.0
3	3	3.000	0.605	$-\frac{16}{9}$	-1.778	0.493	$-\frac{3}{2}$	-1.5
4	$4\cos(\frac{\pi}{7})^2$	3.246	0.873	$-\frac{15}{7} - \frac{3}{28\cos(\frac{\pi}{7})^2}$	-2.275	0.726	-2	-2.0
5	$4\cos(\frac{\pi}{8})^2$	3.414	1.625	$-\frac{21}{8} - \frac{1}{8\cos(\frac{\pi}{8})^2}$	-2.772	1.379	$-\frac{5}{2}$	-2.5
6	$4\cos(\frac{\pi}{9})^2$	3.532	3.782	$-\frac{28}{9} - \frac{5}{36\cos(\frac{\pi}{9})^2}$	-3.268	3.260	-3	-3.0
7	$4\cos(\frac{\pi}{10})^2$	3.618	10.708	$-\frac{18}{5} - \frac{3}{20\cos(\frac{\pi}{10})^2}$	-3.766	9.350	$-\frac{7}{2}$	-3.5

Table 1: The number $c_{k,n}$ ($r_{k,n}$) of compacted trees (respectively relaxed trees) with n internal nodes and right height at most k is asymptotically equal to $\kappa_k n! r_k^n n^{\alpha_k}$ (resp. $\gamma_k n! r_k^n n^{\beta_k}$) with $r_k = \rho_k^{-1}$.

Due to their superexponential growth of order $\Theta(n!\rho_k^{-n} n^{\alpha_k})$ with $\rho_k \approx 1/4$, exponential generating functions are the natural choice. Our second main contribution is the development of a calculus on such exponential generating functions modeling the structural properties of compacted trees in Section 6.

Resulting from these ideas, we were able to give our last main result: the derivation of ordinary differential equations for relaxed and compacted binary trees of bounded right height. The right height of a tree is the maximal number of right edges from the root to any leaf. Furthermore, we extracted the asymptotics by extending the theory of coefficient extractions of ordinary differential equations with polynomial coefficients in Theorem 7.14. This yielded the sought asymptotics in Theorems 3.3 and 3.4.

Thereby we discovered quite *exotic* families of enriched trees. The radii of convergence are in both cases algebraic numbers, and in the case of compacted trees, also the critical exponents are (compare Table 1 for the first 7 families). Note that our techniques do not directly give access to the constants κ_k and γ_k. They can be numerically computed for any specific case from the respective differential equations from the basis of asymptotic solutions like given on page 32 at the end of Section 7. For more details see [25].

It remains an open problem to find the asymptotics of relaxed and compacted trees without any restrictions. For our methods it was crucial that the right height was bounded by a fixed value k. The limit $k \to \infty$ is therefore not computable. In particular, we showed that the radius of convergence ρ_k converges to $1/4$. But the subexponential growth is of the shape $n^{-\lambda k}$ for $\lambda > 0$. Thus, it would converge to 0. Hence, the limits $n \to \infty$ and $k \to \infty$ are not interchangeable. However, in Corollary 5.5 we were able to show that the exponential growth of the number of relaxed and compacted binary trees is equal to 4^n. This behavior remains a topic of future research.

Finally, it is interesting to compare the number of compacted trees to the number of relaxed trees in Corollary 3.5. We showed that their number is negligible for large n and derived a precise quantitative result.

Many new questions arise after our analysis. It would be interesting to consider parameters such as their average height or average right height. Furthermore, these results gave us generating functions of a large family of DAGs which should allow a uniform random generation of such trees. Such results are interesting in computer science and the analysis of algorithms, as DAGs are efficient data structures and widely-used. Among other things, new algorithms need to be tested on very large and non-trivial elements of an efficiently computable class of DAGs.

Acknowledgments

We would like to thank Christian Krattenthaler for pointing out the connection with the Chebyshev polynomials, and Marc Mezzarobba for computing the numerical values of the first few constants κ_k and γ_k to high precision. The authors also thank the anonymous referees for their comments and suggested improvements.

References

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables*, volume 55 of *National Bureau of Standards Applied Mathematics Series*. For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C., 1964.

[2] A. V. Aho, R. Sethi, and J. D. Ullman. *Compilers, Principles, Techniques.* Addison-Wesley, Boston, 1986.

[3] C. Akkan, A. Drexl, and A. Kimms. Generating two-terminal directed acyclic graphs with a given complexity index by constraint logic programming. *J. Log. Algebr. Program.*, 62(1):1–39, 2005.

[4] E. A. Bender, L. B. Richmond, R. W. Robinson, and N. C. Wormald. The asymptotic number of acyclic digraphs. I. *Combinatorica*, 6(1):15–22, 1986.

[5] E. A. Bender and R. W. Robinson. The asymptotic number of acyclic digraphs. II. *J. Combin. Theory Ser. B*, 44(3):363–369, 1988.

[6] O. Bodini, M. Dien, A. Genitrini, and F. Peschanski. The ordered and colored products in analytic combinatorics: Application to the quantitative study of synchronizations in concurrent processes. In *Proceedings of the Meeting on Analytic Algorithmics and Combinatorics*, ANALCO'17, 2017.

[7] O. Bodini, D. Gardy, and B. Gittenberger. Lambda terms of bounded unary height. In *Proceedings of the Meeting on Analytic Algorithmics and Combinatorics*, ANALCO '11, pages 23–32, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics.

[8] O. Bodini, D. Gardy, B. Gittenberger, and Z. Gołębiewski. On the number of unary-binary tree-like structures with restrictions on the unary height. *Ann. Comb.*, 22(1):45–91, 2018.

[9] O. Bodini, D. Gardy, B. Gittenberger, and A. Jacquot. Enumeration of generalized BCI lambda-terms. *Electr. J. Comb.*, 20(4):P30, 2013.

[10] M. Bousquet-Mélou, M. Lohrey, S. Maneth, and E. Noeth. XML compression via directed acyclic graphs. *Theory of Computing Systems*, 57(4):1322–1371, 2015.

[11] V. Carnino and S. De Felice. Sampling different kinds of acyclic automata using Markov chains. *Theoret. Comput. Sci.*, 450:31–42, 2012.

[12] *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.0.13 of 2016-09-16. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller and B. V. Saunders, eds.

[13] P. J. Downey, R. Sethi, and R. E. Tarjan. Variations on the common subexpression problem. *J. Assoc. Comput. Mach.*, 27(4):758–771, 1980.

[14] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

[15] P. Flajolet, P. Sipala, and J.-M. Steyaert. Analytic variations on the common subexpression problem. In *Automata, languages and programming (Coventry, 1990)*, volume 443 of *Lecture Notes in Comput. Sci.*, pages 220–234. Springer, New York, 1990.

[16] P. J. Grabner and B. Steinsky. Asymptotic behaviour of the poles of a special generating function for acyclic digraphs. *Aequationes Math.*, 70(3):268–278, 2005.

[17] P. Henrici. *Applied and Computational Complex Analysis. Vol. 2*. Wiley Classics Library. John Wiley & Sons, Inc., New York, 1991. Special functions—integral transforms—asymptotics—continued fractions, Reprint of the 1977 original, A Wiley-Interscience Publication.

[18] E. L. Ince. *Ordinary Differential Equations*. Dover Publications, New York, 1944.

[19] M. Kauers and P. Paule. *The Concrete Tetrahedron*. Texts and Monographs in Symbolic Computation. SpringerWienNewYork, Vienna, 2011. Symbolic sums, recurrence equations, generating functions, asymptotic estimates.

[20] V. A. Liskovets. Exact enumeration of acyclic deterministic automata. *Discrete Applied Mathematics*, 154(3):537–551, 2006.

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

[22] B. D. McKay. On the shape of a random acyclic digraph. *Math. Proc. Cambridge Philos. Soc.*, 106(3):459–465, 1989.

[23] G. Melançon, I. Dutour, and M. Bousquet-Mélou. Random generation of directed acyclic graphs. In *Comb01—Euroconference on Combinatorics, Graph Theory and Applications*, volume 10 of *Electron. Notes Discrete Math.*, page 6 pp. (electronic). Elsevier, Amsterdam, 2001.

[24] G. Melançon and F. Philippe. Generating connected acyclic digraphs uniformly at random. *Inform. Process. Lett.*, 90(4):209–213, 2004.

[25] M. Mezzarobba. *Autour de l'évaluation numérique des fonctions D-finies.* PhD thesis, École polytechnique, France, Palaiseau, November 2011.

[26] G. Pólya. Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Mathematica*, 68(1):145–254, 1937.

[27] D. Ralaivaosaona and S. Wagner. Repeated fringe subtrees in random rooted trees. In *2015 Proceedings of the Twelfth Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, pages 78–88. SIAM, Philadelphia, PA, 2015.

[28] R. W. Robinson. Enumeration of acyclic digraphs. In *Proc. Second Chapel Hill Conf. on Combinatorial Mathematics and its Applications (Univ. North Carolina, Chapel Hill, N.C., 1970)*, pages 391–399. Univ. North Carolina, Chapel Hill, N.C., 1970.

[29] R. W. Robinson. Counting labeled acyclic digraphs. In *New directions in the theory of graphs (Proc. Third Ann Arbor Conf., Univ. Michigan, Ann Arbor, Mich., 1971)*, pages 239–273. Academic Press, New York, 1973.

[30] R. W. Robinson. Counting unlabeled acyclic digraphs. In *Combinatorial Mathematics, V (Proc. Fifth Austral. Conf., Roy. Melbourne Inst. Tech., Melbourne, 1976)*, pages 28–43. Lecture Notes in Math., Vol. 622. Springer, Berlin, 1977.

[31] L. Schlesinger. *Handbuch der Theorie der linearen Differentialgleichungen. In zwei Bänden, Band I.* Reprint. Bibliotheca Mathematica Teubneriana, Band 30. Johnson Reprint Corp., New York-London, 1968.

[32] R. P. Stanley. *Enumerative Combinatorics. Vol. 2*, volume 62 of *Cambridge Studies in Advanced Mathematics.* Cambridge University Press, Cambridge, 1999. With a foreword by Gian-Carlo Rota and appendix 1 by Sergey Fomin.

[33] B. Steinsky. Enumeration of labelled chain graphs and labelled essential directed acyclic graphs. *Discrete Math.*, 270(1-3):267–278, 2003.

[34] B. Steinsky. Asymptotic behaviour of the number of labelled essential acyclic digraphs and labelled chain graphs. *Graphs Combin.*, 20(3):399–411, 2004.

[35] S. Wagner. Asymptotic enumeration of extensional acyclic digraphs. *Algorithmica*, 66(4):829–847, 2013.

[36] M. Wallner. A bijection of plane increasing trees with relaxed binary trees of right height at most one. *Theoretical Computer Science*, 755:1–12, 2019.

[37] W. Wasow. *Asymptotic Expansions for Ordinary Differential Equations.* Dover Publications, Inc., New York, 1987. Reprint of the 1976 edition.