Combinatoire des arbres sous étiquetages croissants: Asymptotiques, bijections et algorithmes

Combinatorics of trees under increasing labellings:

Asymptotics, bijections and algorithms

Mehdi Naima

Sous la direction de:
Olivier Bodini et Antoine Genitrini

THÈSE
pour obtenir le titre de
Docteur en Sciences mention Informatique
École doctorale Galilée (ED 146)
Université Sorbonne Paris Nord
France

Rapporteur:

Julien Clément	Université de Caen
Alois Panholzer	Technische Universität Wien

Jury:

Frédérique Bassino	Université Sorbonne Paris Nord
Olivier Bodini	Université Sorbonne Paris Nord
Julien Clément	Université de Caen
Antoine Genitrini	Sorbonne Université
Cécile Mailler	University of Bath
Cyril Nicaud	Université Gustave Eiffel
Alois Panholzer	Technische Universität Wien
Vlady Ravelomanana	Université de Paris

18 Décembre 2020

Résumé

Dans cette thèse nous étudions des classes d'arbres étiquetés selon différents modèles d'étiquetages croissants. Ces arbres sont utiles dans la modélisation de nombreux processus.

Nous adoptons dans nos recherches différents points de vues complémentaires (combinatoire, probabiliste ou informatique) afin d'enrichir les résultats connus sur les arbres croissants classiques et de proposer de nouvelles classes d'arbres moins contraints que les modèles existants dans la littérature.

Nous proposons plusieurs nouveaux modèles d'arbres, dans l'idée de pouvoir représenter un processus d'évolution où l'historique des évolutions est enregistré. Pour ces nouvelles classes d'arbres nous montrons leur liens étroits avec des objets classiques en combinatoire comme les permutations, les partitions d'ensemble, ainsi que les graphes. Nous les étudions également de façon plus détaillée en terme probabiliste pour mieux comprendre la forme typique des grandes structures.

Ainsi, nous définissons un processus d'évolution paramétrable qui recouvre ces nouvelles classes d'arbres ainsi que d'autres classes encore plus générales. Cela nous mène à définir plusieurs nouveaux modèles d'étiquetages croissants sur les arbres. Nous réussissons aussi à avoir des formes universelles pour l'énumération asymptotique des classes d'arbres issues de ce processus d'évolution en utilisant notamment des idées empruntées aux sommations de Borel.

Du côté algorithmique l'étude des structures arborescentes nécessite la génération et la mémorisation d'arbres de grandes taille ce qui nous mène à élaborer des algorithmes de génération aléatoire uniforme efficaces qui nous permettent de faire des simulations non biaisées sur des arbres de grandes taille. Du fait que nous sommes en mesure d'engendrer des arbres grands, une autre problématique apparaît. Celle-ci concerne leur représentation en mémoire. En particulier, nous nous sommes aperçus qu'une compression efficace serait intéressante afin de manipuler et étudier expérimentalement ces structures arborescentes. Notre étude porte alors sur le taux compression moyen des arbres croissants classiques et elle nous permet de définir une nouvelle structure de données compactifiée pour les arbres binaires de recherche.

Abstract

In this thesis we study classes of trees labelled according to different increasing labellings. These trees are useful in the modelisation of various processes.

In our research different we adopt different but complementary points of view (combinatorial, probabilistic or algorithmic) to enrich the results on classical increasing trees and propose new tree classes that are less constrained than existing models in the literature.

We propose several new tree models, which each represent a process of evolution where the histories of evolutions are registered. For these new classes of trees we show their close links with classic combinatorial objects such as permutations, set partitions and graphs. We also study them in more detail in probabilistic terms to have a better understanding of the typical shape of large structures.

Then, we define a parametrizable evolution process that covers these new classes of trees as well as other even more general classes. This leads us to define several new models of increasing labellings on trees. We also derive universal forms for the asymptotic enumeration of tree of classes of trees resulting from this evolution process using, in particular, ideas borrowed from Borel's summations.

On the algorithmic side, the study of tree structures requires the generation and storage of large trees. This leads us to develop uniform random generation algorithms which allow us to make unbiased simulations on large trees. Since we are able to generate large trees, another problem arises concerning their representation in memory. In particular, we have found that efficient compression would be useful for manipulating and studying experimentally these tree structures. Our study then focuses on the average compression ratio of increasing trees and it allows us to define a new compacted representation of binary search trees.

Remerciements

> کی اپ اکِینون اِن کالی زوی مَس[1].
>
> *Rumi (1207 - 1273), Diwan-e*
> *Shams-e Tabrizi*[2]

Bien que ce document porte mon nom il m'aurait été impossible de le produire seul, chaque personne ayant de près ou de loin participé mériterait d'y figurer. c'est en gardant ce fait à l'esprit que j'adresse mes sincères remerciements.

Il serait plus simple pour moi de dire ce que mes directeurs Olivier Bodini et Antoine Genitrini n'ont pas fait. Ils n'ont pas contribué à l'écriture de ces remerciements. À part cela, leur bienveillance, leur soutien permanent et leur grande gentillesse sont présents dans chaque pas de mes recherches ainsi que dans ceux de ma formation académique. Sans eux, je n'aurai rien fait de tout cela. [3]

Je souhaite aussi remercier de tout coeur les rapporteurs de ma thèse Alois Panholzer et Julien Clément pour leurs remarques bienveillantes et constructives afin d'améliorer mon manuscrit de thèse. Et plus spécialement Julien Clément pour tout le temps qu'il m'a consacré depuis le début de ma thèse, sa bonne humeur et sa gentillesse à chacune de nos rencontres.

Je suis très reconnaissant envers les membres du jury de ma thèse; Frédérique Bassino, Julien Clément, Cécile Mailler, Cyril Nicaud, Vlady Ravelmonana et mes directeurs, d'avoir accepté ce rôle. Je suis honoré de leur présence. Et suis particulièrement reconnaissant envers Frédérique Bassino en tant que directrice du Laboratoire Informatique de Paris Nord pour sa disponibilité et son aide durant ma thèse et pour la préparation de la soutenance.

Un grand merci à tous mes coauteurs qui m'ont appris énormément de choses. Je serai très heureux de continuer à collaborer avec chacun d'entre eux à l'avenir. J'ai été très touché par l'accueil chaleureux et la douceur de Cécile Mailler lors de mon voyage à Bath ainsi que pour les méthodes probabilistes que j'ai appris en la côtoyant. Merci à Isabella Larcher pour tous ces moments partagés et ces bonnes conversations à Paris et à Vienne sur nos vies et sur les arbres "croissants"; À Bernhard Gittenberger pour son accueil à Vienne, sa grande

[2] English translation : 'It is because of him that life is sweet'. Rumi wrote it as a Greek Phrase in the Persian script. The Greek equivalent is 'κι απ' εκείνον έν καλή η ζωή μας'.

[2] A collection of lyric poems that is considered one of the great works in Persian literature.

[3] Je reformule ici les remerciements qu'a rédigé Daniel Greene à son directeur Donald Knuth. À bien des égards ces deux chercheurs sont des parents académiques pour moi.

gentillesse et toutes les techniques de combinatoire et d'analyse complexe que j'ai appris avec lui. Sans oublier Alexandros Singh qui m'apprend chaque jour de nouvelles choses et qui a participé grandement à l'évolution de mes travaux, qui s'est toujours montré présent. Sa sérénité, sa bonne humeur et son amitié précieuse m'apporte beaucoup.

Je souhaite à présent remercier ma famille doctorale en commençant par Matthieu Dien pour son accueil lors de mes voyages à Caen et à Taïwan, ainsi que pour sa patience, son sens de l'humour et son attention. Toute ma reconnaissance se porte ensuite envers Martin Pépin pour son aide scientifique et pour nos voyages en Autriche et à Marseille. C'est un formidable compagnon. Elle se porte également envers Sergey Dovgal, pour son énergie scientifique permanente et notre randonnée dans les calanques.

Le travail sur ma thèse m'a amené à faire de merveilleuse rencontres. Andrew Elevy Price en fait parti. Je tiens à souligner ici la joie que j'ai de partager chacun de nos moments ensemble, nos longues déambulations dans les rues de Paris sans oublier nos parties de jeux de société où je perds presque toujours. Je lui suis aussi reconnaissant pour tout ce qu'il m'a appris en combinatoire et bien au-delà.

Merci aussi à Michael Wallner pour son invitation à Bordeaux et tous les bons moments que j'ai passé en sa compagnie. J'ai rarement vu une personne aussi souriante et bienveillante.

Je suis aussi reconnaissant envers Julien David pour m'avoir introduit au monde de la recherche ainsi que pour toutes nos discussions et balades dans Paris.

Rencontrer Frédérique Peschanski, Hsien-Kuei Hwang, Cyril Banderier, Mireille Bousquet-Mélou, Gérard Duchamp, Pierre Nicodème, Andrea Sportiello, Julien Courtiel, Danièle Gardy, Noam Zeilberger, Clément Requilé, Elie de Panafieu, Gérard Duchamp et Thierry Monteil fut pour moi également une grande joie. je tiens à tous les remercier.

Comme nos recherches ne seraient rien sans le travail de nos prédécesseurs il me parait indispensable de remercier tous les savants dont les travaux m'ont été bénéfiques et particulièrement Philippe Flajolet.

Le bureau dans lequel j'ai travaillé était rempli de bonne humeur grâce aux doctorants du LIPN avec qui je l'ai partagé. Davide Barbarossa, Jawher Jerray, Le Thanh Dung Nguyen (Tito), Enrico Bettiol, Alexandros Singh, Sarah Zouinina, Tiphaine Viard ,Dina Faneva, Ugochukwu Onwudebelu et Juan José Figueroa, ont toujours su m'encourager et m'épauler. Il en fut de même des doctorants de l'équipe APR du LIP6 Clément Bertrand, Raphaël Monat, Ghiles Ziat et Martin Pépin.

Je souhaite aussi remercier les écoles que j'ai fréquentées et qui ont aidées à ma formation académique. La mosquée Al-Shafii, École Omar ibn al-Khattab, l'École al-Ibaa al Araby, l'École al-Mohsiniyyah, l'Université privée de Yarmouk, l'Université de Sorbonne Paris Nord, Sorbonne Université, l'Université de Montréal ainsi que l'université de Paris. Je souhaiterais aussi remercier tous mes camarades proches qui ont aidé et contribué bien suivre les cours ainsi qu'à l'ensemble de mes professeurs.

Mes remerciements vont également à toutes les personnes qui contribuent à développer des logiciels qui m'ont été d'une grande aide durant mes années de thèse, notamment l'excellente base données OEIS et son fondateur Neil Sloane ainsi que LaTeX, TeX, Emacs, Sagemath, Wolfram, Maple, Graphwiz.

De façon plus générale je remercie toutes les personnes de la communauté Analysis of Algorithms (AofA) et ceux d'ALEA.

Milles mercis à mon ami Tarek Hussni à qui je dois entre autres la reprise de mes études universitaires. Je le remercie surtout mon pour son amitié et sa bienveillance ainsi qu'à sa famille.

Je remercie aussi mes autres amis proches qui m'ont toujours encouragé, Saer Alkhyat, Ayman AlHajjeh, Anas Shekfa, Yannis Bensalah, Jean Peyen, Arsalan Riaz, Omar Altinawi, Osmana Traore, Saad Nahlawi, Abdulrahman Hamdan et la famille Schmitz.

Je remercie de même mes cousins français et syriens pour leur sympathie et tous les bons moments que l'on partage et également mes oncles et tantes ainsi que mes grands-parents. J'ai une pensée particulière pour mes cousins et amis emprisonnés en Syrie depuis dans années.

Merci également à mes soeurs pour leur tendresse et leur douceur, et à mon petit frère pour son courage et toute l'énergie qu'il véhicule.

J'ai une pensée très forte à Brigitte Henry, qui a toujours été un grand soutien et une inspiration précieuse pour moi. Je lui suis profondément reconnaissant. Qu'elle repose en paix.

Je remercie également Magali Richard, pour son amour, sa bonté ainsi que pour son aide permanente et notamment pour le beau dessin sur la page de dédicaces.

Pour conclure je remercie mes parents à qui je dois tout. Mon père passe sa vie au travail pour que l'on ne manque jamais de rien et ma mère a toujours tout sacrifié pour notre bien-être. Merci pour tout.

La liste est longue des gens qui comptent pour moi et qui ont contribué à sans nécessairement s'en rendre compte à ce travail. Par avance désolé si un nom viendrait à manquer. Il ne fait nul doute que chacun est important pour moi.

De tout coeur merci à tous

Contents

List of Symbols

$\left[{n \atop k}\right]$	Stirling cycle numbers (First kind)
$\left\{{n \atop k}\right\}$	Stirling partition numbers (Second kind)
$\mathcal{S}_n$	The set of all permutations of n elements
$\left\langle{n \atop k}\right\rangle$	The number of permutations of size n with k descents
$\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots$	Combinatorial classes
$\mathcal{A}_n, \mathcal{B}_n, \mathcal{C}_n, \ldots$	Class of objects of $\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots$ which have size n
$A_n, B_n, C_n, \ldots$	Number of objects of size n in $\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots$
$A(z), B(z), C(z), \ldots$	Generating function of $\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots$
$n^{\underline{k}}$	$n\,(n-1)\;\ldots,(n-k+1)$
$[n]$	$\{1, 2, \ldots, n\}$

À mes parents,
À Brigitte Henry,
Et à toutes les victimes de la tyrannie,

CHAPTER 1

Introduction

> Trees are poems that the earth writes upon the sky.
>
> *Kahlil Gibran (1883 - 1931),*
> *Sand and Foam.*

Tree structures are widely known and commonly studied objects which applications in various fields and disciplines ranging from computer science and mathematics to biology, phylogenetics and sociological research.

In computer science, some examples of trees include tree structures that are used as representations of the abstract syntactic structure of source code written in a programming language. In this context they are called *abstract syntax trees*. In computational linguistics, a *parse tree* represents the syntactic structure of a string according to some context-free grammar. *Parse trees* together with *abstract syntax trees* are used as main steps in the compiling process of a program. See [**ALSU06, Muc97, GvRB^{+}12**] for some references on the subject.

Other examples include, markup languages such as XML that have underlying tree structures that can be used and processed subsequently by the *Document Object model*, see [**HM04**] for an introduction. Finally, in data structures, trees are used as efficient structures to store and extract information. The most famous examples are *binary search trees*, *red-black trees* and *AVL trees*, some references include [**CCLR09, RK11**].

On one hand, tree structures are naturally defined as subclasses of *directed acyclic graphs* (or DAGs) that are very common in graph theory because they can model many different kinds of information and they are in direct relation with partial orders as well. On the other hand *directed acyclic graphs* appear naturally in computer science in the context of tree-structures that are compacted by sharing substructures. Compression of data structures is not only studied computer science where it is a central tool in order to save memory, but is also important in different fields such as information theory and combinatorics. There, it is related to the central concepts of symmetries, entropy and Shannon information, see [**CT05**] for an overview on information theory and compaction of data structures.

Trees can also be labelled, such that each node of the tree contain some label. We have already mentioned trees used in data structures, in which we manipulate data. In combinatorics, people are interested in many different kinds of tree labellings. Some books on the subject include [**CCG18**, **FS09**]. An important case of labellings is the *increasing labelling*, where the labels in the nodes of the tree are increasing along branches.

Some models include *increasing trees* which have been introduced by Bergeron, Flajolet and Salvy in [**BFS92**], where the authors study trees with no label repetitions. Other models include [**PU83**], where the authors study increasing trees with label repetitions and more recently [**BGGW20**]. Label repetitions in concurrency theory can represent synchronisation of processes as in [**Gen17**]. Other studies include [**KP16**] in which the authors look at increasing trees that are multi-labelled, that is, a node can contain several labels.

This kind of notion is adapted to the study of several tree classes like binary search trees that are equivalent to *increasing binary trees* as well as recursive trees, and plane oriented (or heap ordered) trees, and monotonic trees, see [**Drm09**].

Increasing trees are adapted to the study and analysis of dynamic evolution processes, such that phylogenetic trees that represent the evolutionary relationship among species. At each bifurcation (or multifurcation) of the tree, the descendant species from distinct branches have differentiated. Increasing trees can usually described as an *incremental process* where nodes are inserted at different iteration steps according to some distribution as in [**PP07**]. This models the fact that at any given time, each existing species is equally likely to give rise to new species. More information on trees in biology and phylogenetics can be found in [**Fel03**, **Ste16**]. These tree models can also be used as simple models for epidemics or other evolution processes that admit incremental modelisation.

In this thesis we study various classes of trees under different increasing labellings. Our perspective, varies from that of a combinatorialist, with questions of enumeration, bijections and asymptotics, to the one of a probabilist, with analysis questions and determination of limit random laws, to that of a computer scientist, where questions relating algorithm designs, data structures and random generation are dealt with.

These different points of views, allow us to make rigorous theoretical statements, about the objects of study and then design practical algorithms that manipulate these objects.

This thesis consists of three main parts. First, we introduce three new models of increasing trees, which are suitably interpreted as evolutionary trees, but they are also adapted to represent other evolution processes such as that of programs in concurrency theory.

The underlying structure of these tree models is the one of *Schröder trees* that were studied by Ernst Schröder [**Sch70**] in connection with evolutionary trees. However, his framework did not model the of new species (i.e differentiations) over time, which we do by considering labellings on internal nodes. These trees are counted by their number of leaves and internal nodes bear integer labels. The first model called *increasing Schröder trees* only allows for one species to evolve at a fixed period of time, while the other two models *strict monotonic Schröder trees* and *strict monotonic general Schröder trees*, allow for several species to evolve simultaneously.

For all three models, we will also study the average value of some parameters, in order to have a better understanding of the typical shape of large random trees that belong to this classes. We will also see how these models relate to classical objects in combinatorics such as *permutations*, *set partitions* and *graphs*. In particular, we will exhibit relations between cycles in permutations and the number of internal nodes or the depth of a fixed leaf. We will also see how *Eulerian numbers* and *Stirling numbers* of both kinds relate on the tree structure.

The second main part of our research lies in the presentation of a general parameterisable evolution process for classes of *increasing Schröder trees* that encompasses all three models already presented but also many others. This will also allow us to introduce *weakly increasing labellings* on tree structures in a new way that is different from [**PU83**]. For this general evolution process, we will be mainly concerned with the asymptotic enumeration of these classes of trees. Our theorems give general asymptotic formulae that are in the same spirit as the theorem for universal asymptotic behaviour of simply generated trees in [**FS09**].

On a theoretical level, we see that the labellings that we add to the tree structures which allows for repeated labelling and weak increasing labelling along branches are easily specified using *ordinary generating functions*. However, the generating functions are then invariably divergent. Therefore, we develop a general method which is related to Borel summations, which allows to capture the asymptotic behaviour.

Finally, the last axis of our research concerns data compression. We will study the average compaction rate of trees under the *increasing labelling distribution*. Our aim is to extend already known results on the average compaction of trees under the *uniform distribution* that has been studied under the name of "common subexpression recognition" in [**FSS90**] and more recently by [**BMLMN15**]. For instance, we will show that the average compaction of binary trees under the *increasing labelling distribution* is better than the average compaction under *uniform distribution*. By the word 'better' here, we mean that we gain an asymptotic order in the average compaction rate. In light of our theoretical results on the compaction of binary trees, we will propose a new lossless data structure based on the compaction of binary search trees. In the same way we study *Pólya trees* under the *increasing labelling distribution*.

Plan of the thesis:

We start in Chapter 2 by presenting the main theoretical tools that we will use to build our results. Chapter 3 contains a presentation of the different known combinatorial objects that are used throughout the thesis. Chapter 4 introduces the three new classes of increasing Schröder trees, it contains a thorough study of these three classes in terms of enumeration, asymptotics, typical shapes, and relationship with classical combinatorial objects. In Chapter 5, we present a general evolution process that includes the three models of increasing Schröder trees as parametrisations. This chapter is mainly concerned with the asymptotic enumeration of the different classes of trees that can be produced with the evolution process. Then, Chapter 6 is dedicated to the study of average tree compaction of two tree models under *increasing labelling distribution*. Finally, in Chapter 7, we talk about the *uniform random generation* of classes of trees that are the result of the evolution process defined in Chapter 5.

We also show how the three classes of increasing Schröder trees defined in Chapter 4 admit efficient uniform sampling algorithms.

O. Bodini, A. Genitrini, M. N.	“Ranked Schröder trees”. In Proceedings of the Sixteenth Workshop on Analytic Algorithmics and Combinatorics (ANALCO), 2019.	Chapter 4
O. Bodini, A. Genitrini, C. Mailler, M. N.	“Strict monotonic trees arising from evolutionary processes: combinatorial and probabilistic study”. Submitted to a journal. Available on https://hal.sorbonne-universite.fr/hal-02865198	Chapter 4, Chapter 7
O. Bodini, A. Genitrini, M. N., A. Singh	“Families of Monotonic Trees: Combinatorial Enumeration and Asymptotics”. In Proceedings of the 15th International Computer Science Symposium in Russia (CSR), 2020.	Chapter 5
O. Bodini, A. Genitrini, B. Gittenberger, I. Larcher, M. N.	“Compaction for two models of logarithmic-depth trees: Analysis and Experiments”. Submitted to a journal. Available on https://arxiv.org/abs/2005.12997	Chapter 6

Table 1.1: Publications and preprints that form parts of this thesis.

CHAPTER 2

Methods

Contents

> Ainsi mon dessein n'est pas d'enseigner la méthode que chacun doit suivre pour bien conduire sa raison, mais de faire voir en quelle sorte j'ai tâché de conduire la mienne.[1]
>
> *René Descartes (1596 - 1650),*
> *Discours de la méthode.*

[1] Thus my design is not here to teach the Method which everyone should follow in order to promote the good conduct of his Reason, but only to show in what manner I have endeavoured to conduct my own.

This chapter presents briefly the main concepts and mathematical tools that will be used throughout this thesis.

2.1 Power series

Let A be a commutative rings and z an indeterminate. We denote by $A[[z]]$ the ring of formal power series in z with coefficients in A.

Unless otherwise stated we will work with power series in $\mathbb{C}[[z]]$ with elements of the form:

$$A(z) = \sum_{n \geq 0} A_n z^n.$$

The function $A(z)$ is then an element of $A[[z]]$. Power series are useful in analysis especially as Taylor series. Émile Borel a french mathematician showed that every power series is the Taylor series of some smooth function. For example, let for all $n \geq 1, A_n = 1, B_n = \frac{1}{n!}$ then,

$$A(z) = \frac{1}{1-z}, \qquad B(z) = \exp(z).$$

Power series are used as a basis for generating functions in combinatorics.

Depending on the values of the indeterminate z the function may converge or diverge. There is always a number $R \geq 0$, such that the power series converges for all values $z < R$ and diverges when $z > r$. This value R is called the *radius of convergence* of the power series.

As it will be seen in Section 2.4, the *radius of convergence* plays an important role in determining the asymptotic behaviour of a combinatorial class.

Power series behave nicely with algebraic operations that can be defined on them to combine them and form new power series. Table 2.1 gives a summary of some of the most basic operations. When defining or combining power series together we consider the indeterminate z as a formal variable without considering the questions of convergence and uniformity. These questions will be inverstigated when looking for asymptotic behaviour of power series seen as counting objects.

A good introduction to the subject with advanced operations on formal power series can be found in the works of Goulden, Jackson and Wilf among others [**GJ04, Wil05**].

Using the definition of a power series a *polynomial* is an element of $A[[z]]$ with only a finite number of nonzero coefficients.

The definition of formal power series can be extended to the multivariate case. Therefore $A[[z, u]]$ has elements of the form:

$$A(z, u) = \sum_{n \geq 0} \sum_{k \geq 0} A_{n,k} u^k z^n$$

[2] In this thesis we will use ∂_z for $\frac{d}{dz}$.

Coefficient extraction	$[z^n]A(z) = A_n$	
Addition	$C(z) = A(z) + B(z)$	$\forall n \geq 0, C_n = A_n + B_n$
Cauchy Product	$C(z) = A(z) \cdot B(z)$	$\forall n \geq 0, C_n = \sum\limits_{k=0}^{n} A_k B_{n-k}$
Derivative	$C(z) = \partial_z A(z)$ [2]	$\forall n \geq 0, C_n = (n+1)A_{n+1}$
Integral	$C(z) = \int\limits_0^z A(z)$	$\forall n \geq 1, C_n = \frac{A_{n-1}}{n}$
Exponential series	$C(z) = \exp(z)$	$\forall n \geq 0, C_n = \frac{1}{n!}$
Logarithmic series	$C(z) = \log(1-z)^{-1}$	$\forall n \geq 1, C_n = \frac{1}{n}$
Binomial series	$C(z) = (1+z)^k$	$\forall n \geq 0, C_n = \binom{k}{n}$

Table 2.1: Basic operations on power series

And we have

$$[z^i] \sum_{n \geq 0} \sum_{k \geq 0} A_{n,k} u^k z^n = \sum_{k \geq 0} A_{i,k} u^k$$

In order to extract both coefficient in the same time we denote $[z^i u^j]A(z,u) = A_{i,j}$.

2.2 Symbolic Methods

This section is mostly based on [FS09, Ch 1 and 2].

Definition 2.2.1. *A combinatorial class set of objects on which a size notion has been defined such that the size s of an object is always positive $s \geq 0$ and the number of objects of a fixed size is finite.*

Symbolic methods are methods that help constructing combinatorial classes. These methods give the correspondence between operations to construct classes of combinatorial objects and their corresponding operations on the generating function level.

Notation. We will denote combinatorial classes with calligraphic capital letters like $\mathcal{A}, \mathcal{B}, \mathcal{C}$, and the notation $\mathcal{A}_n$ will refer to the combinatorial class defined as the subset of elements of $\mathcal{A}$ that have size n.

2.2.1 Ordinary generating functions

Ordinary generating functions are usually used to specify combinatorial objects that are unlabelled. However, in this thesis we will see how they can be used to specify increasing structures.

Definition 2.2.2. *The* ***ordinary generating function*** *(or* ***OGF****) of a sequence A_n is the formal power series*

$$A(z) = \sum_{n \geq 0} A_n z^n.$$

Ordinary generating functions are also referred to simply as *generating functions*. According to Georges Pólya in [**Pól54**] the name *generating functions* was coined by Pierre-Simon Laplace but its use dates back to Leonhard Euler.

We start with the basic operators. The generating function of the neutral class $E(z)$ (contains a single object of size 0) and the atomic class (contains a single object of size 1) $Z(z)$

$$E(z) = 1 \qquad Z(z) = z.$$

The *combinatorial sum* (or disjoint union) between two combinatorial classes, create a new class by putting together all the elements of each class. It might be the case that some element are common to both classes, in which case we colour each element with a new colour and add both to the new class. The size of an element in the new class is inherited from the class it came from.

The *Cartesian product* forms all possible ordered pairs between two classes. The size of an object (which is a pair) is obtained by adding the size of both objects contained in it.

Sum	$\mathcal{A} = \mathcal{B} + \mathcal{C}$	$A(z) = B(z) + C(z)$
Cauchy Product	$\mathcal{A} = \mathcal{B} \times \mathcal{C}$	$A(z) = B(z) \cdot C(z)$
Sequence	$\mathcal{A} = \text{Seq}(\mathcal{B})$	$A(z) = \dfrac{1}{1 - B(z)}$
Pointing	$\mathcal{A} = \Theta(\mathcal{B})$	$A(z) = z\partial_z B(z)$
Substitution	$\mathcal{A} = \mathcal{B} \circ \mathcal{C}$	$A(z) = B(C(z))$
Multiset	$A = \text{MSet}(\mathcal{B})$	$A(z) = \prod_{n \geq 1} (1 - z^n)^{-B_n}$

Table 2.2: Admissible constructions from [**FS09**, Ch 1]

The *Sequence* is defined for a class $\mathcal{A}$ as the infinite sum of

$$\text{Seq}(\mathcal{A}) = \{\epsilon\} + \mathcal{A} + (\mathcal{A} \times \mathcal{A}) + (\mathcal{A} \times \mathcal{A} \times \mathcal{A}) + \dots$$

The *pointing* of a class $\mathcal{A}$ is a new class made by selecting any unit element in $\mathcal{A}$. More formally,

$$\Theta(\mathcal{A}) = \sum_{n\geq 0} \mathcal{A}_n \times \{\epsilon_1, \epsilon_2, \ldots, \epsilon_n\}$$

The *substitution* or (*composition*) is defined by

$$\mathcal{B} \circ \mathcal{C} = \sum_{n\geq 0} \mathcal{B}_n \times \mathrm{Seq}_{\mathrm{k}}(\mathcal{C})$$

The *Multiset* construction can be decomposed into

$$\mathrm{MSet}(B) \cong \prod_{\beta\in\mathcal{B}} \mathrm{Seq}\left(\{\beta\}\right).$$

In fact a multiset can always be reorganised as a product of sequences of each element of $\mathcal{B}$. For instance if $\mathcal{B} = \{1, 2, 3\}$, the element $\{1, 2, 1, 3, 1, 2\}$ can be seen as $\{1, 1, 1, 2, 2, 3\}$. Therefore we can get also an alternative form for the multiset construction,

$$\begin{aligned} A(z) &= \prod_{\beta\in\mathcal{B}} (1 - z^{|\beta|})^{-1} = \prod_{n\geq 1} (1 - z^n)^{-B_n} \\ &= \exp\left(\sum_{n\geq 1} B_n \log\left(1 - z^n\right)^{-1}\right) \\ &= \exp\left(\frac{B(z)}{1} + \frac{B(z^2)}{2} + \frac{B(z^3)}{3} + \ldots\right). \end{aligned}$$

In this section we focus on the description of construction operators but many examples will follow in Chapter 3.

2.2.2 Exponential generating functions

Definition 2.2.3. *The **exponential generating function** (or **EGF**) of a sequence A_n is the formal power series*

$$A(z) = \sum_{n\geq 0} A_n \frac{z^n}{n!}.$$

Exponential generating functions will be denoted for short **EGF**. They are known to be adapted to labelled combinatorial structures.

Definition 2.2.4. *A labelled combinatorial class is a combinatorial class, such that each object is labelled with labels in $\{1, \ldots, n\}$, where n is the total size of the object and the labels are all distinct.*

The neutral and the atomic class objects are the same as in Section 2.2.1 but now the atomic class refer to the only object of size 1, that is labelled 1.

Disjoint sum	$\mathcal{A} = \mathcal{B} + \mathcal{C}$	$A(z) = B(z) + C(z)$
Labelled Product	$\mathcal{A} = \mathcal{B} \star \mathcal{C}$	$A(z) = B(z) \cdot C(z)$
Sequence	$\mathcal{A} = \mathrm{Seq}(\mathcal{B})$	$A(z) = \dfrac{1}{1 - B(z)}$
Set	$\mathcal{A} = \mathrm{Set}(\mathcal{B})$	$A(z) = \exp(B(z))$
Cycle	$\mathcal{A} = \mathrm{Cyc}(\mathcal{B})$	$A(z) = \log \dfrac{1}{1 - B(z)}$
Boxed product	$\mathcal{A} = \mathcal{B}^{\square} \star \mathcal{C}$	$A(z) = \int\limits_0^z (\partial_t B(t)) \cdot C(t)dt$

Table 2.3: Admissible constructions from [FS09, Ch 2]

If we have two labelled objects $\beta \in \mathcal{B}$ and $\gamma \in \mathcal{C}$, their labelled product is written by $\beta \star \gamma$, is a set consisting of well-labelled ordered pairs (β', γ') that reduce to (β, γ). If β has size i and γ size j. If $n = i + j$, then the number of elements in $\beta \star \gamma$ is $\binom{n}{i}$.

The *labelled product* of two classes $\mathcal{B}$ and $\mathcal{C}$, denoted $\mathcal{B} \star \mathcal{C}$, is then obtained by forming ordered pairs from $\mathcal{B} \times \mathcal{C}$ and performing all possible order-consistent relabellings. We have,

$$\mathcal{B} \star \mathcal{C} = \bigcup_{\beta \in \mathcal{B}, \gamma \in \mathcal{C}} (\beta \star \gamma).$$

The *sequence* is defined for a class $\mathcal{A}$ as before

$$\mathrm{Seq}(\mathcal{A}) = \{\epsilon\} + \mathcal{A} + (\mathcal{A} \star \mathcal{A}) + (\mathcal{A} \star \mathcal{A} \star \mathcal{A}) + \dots$$

We can define a k-*sequence*,

$$\underset{k}{\mathrm{Seq}}(\mathcal{A}) = \underbrace{\mathcal{A} \star \cdots \star \mathcal{A}}_{k}$$

A *sequence* is also

$$\mathrm{Seq}(\mathcal{A}) = \bigcup_{k \geq 0} \underset{k}{\mathrm{Seq}}(\mathcal{A}).$$

From this a k-*set*, denoted by $\mathrm{Set}_k(\mathcal{A})$, corresponds to the quotient class of elements of $\mathrm{Seq}_k(\mathcal{A})/R$, where R is the equivalence relation that identifies two sequences when the components of one are a permutation of the components of the other. From here the set class is simply

$$\mathrm{Set}(\mathcal{A}) = \bigcup_{k \geq 0} \underset{k}{\mathrm{Set}}(\mathcal{A}).$$

In the same spirit, we can define a k-*cycle*, denoted by $\mathrm{Cyc}_k(\mathcal{A})$, corresponding to the quotient class of elements of $\mathrm{Seq}_k(\mathcal{A})/S$, where S is the equivalence relation that identifies two sequences when the components of one are a cyclic permutation of the components of the other. From here the cycle class is simply

$$\mathrm{Cyc}(\mathcal{A}) = \bigcup_{k\geq 0} \mathrm{Cyc}_k(\mathcal{A}).$$

The boxed (or Greene) product is used for order constraint which is very useful for increasing structures. It has been introduced by Greene in his thesis [Gre91]. It is defined as the subset of the product $\mathcal{B} \star C$ such that the smallest label is constrained to in the $\mathcal{B}$ component. For consistency we also need to have $B_0 = 0$. We get for its coefficients

$$A_n = \sum_{k=1}^{n} \binom{n-1}{k-1} B_k C_{n-k}$$

Remark 2.2.5. *A simple relation can be deduced directly between a sequence and a set of cycles.*

$$\mathrm{Seq}(A) = \mathrm{Set}(\mathrm{Cyc}(A)).$$

Since,

$$\frac{1}{1-A(z)} = \exp\left(\log\left(\frac{1}{1-A(z)}\right)\right).$$

A combinatorial example of this fact can be seen on permutations in Section 3.1.

2.3 Combinatorial Borel and Laplace transforms

These two transforms are known to be 'bridges' between **OGF** and **EGF**. Since they transform a generating function from one type to the other. The combinatorial Borel transform is defined by

$$\mathcal{B}\sum_{n\geq 0} a_n z^n = \sum_{n\geq 0} a_n \frac{z^n}{n!},$$

and the combinatorial Laplace,

$$\mathcal{L}\sum_{n\geq 0} a_n \frac{z^n}{n!} = \sum_{n\geq 0} a_n z^n.$$

Therefore the ***Laplace transform*** converts an **EGF** to an **OGF** while the ***Borel transform*** does the inverse. The Laplace transform can be analytic under suitable conditions of convergence and can be defined as:

$$\mathcal{L}f(z) = \int_0^\infty f(zx)\, \mathrm{e}^{-x}\, dx.$$

While the Borel transform can be defined by:

$$\mathcal{B}f(z) = \frac{1}{2i\pi} \int_{c-i\infty}^{c+i\infty} \frac{\mathrm{e}^{zt}}{t} f\left(\frac{1}{t}\right) dt,$$

where c is greater than the real part of all singularities of $f(\frac{1}{t})\, t^{-1}$.

Some simple rules can be directly inferred from these definitions:

- $\mathcal{L}f' = \dfrac{1}{z}(\mathcal{L}f - f_0)$.
- $\mathcal{L}(\int f) = z\mathcal{L}f$.
- $\mathcal{B}(zf) = \int f$.
- $\mathcal{B}(\dfrac{f - f_0}{z}) = (\mathcal{B}f)'$.
- $\mathcal{B}f' = (\mathcal{B}f)' + z(\mathcal{B}f)''$.

For more information on these transforms and on new operators for symbolic methods see [**BDGP17**] in which the authors introduce new operators for the symbolic method. More detailed accounts are also found in the thesis of [**Die17**].

These transforms and especially the Borel one, will be very useful to us in the subsequent chapters. This is due to the fact that many of our specifications will made in the unlabelled world while the objects of study will belong to the labelled one. Therefore, we will be passing from **OGF**s to **EGF**s in order to compute asymptotics.

Combinatorial Borel transforms are closely related to Borel summation with the idea of summing divergent power series introduced by Émile Borel in 1899.

2.4 Analytic methods for asymptotics

Most of the material of this section is taken from [**FS09**, ch 4,6,7]. We start this section with a reminder of Cauchy's coefficients formula.

Theorem 2.4.1. *Let f (z) be analytic in a region Ω containing 0 and let λ be a simple loop around 0 in Ω that is positively oriented. Then, the coefficient $f_n = [z^n]f(z)$ admits the integral representation*

$$f_n = \frac{1}{2i\pi} \int\limits_{\gamma} f(z) \frac{dz}{z^{n+1}}.$$

This formula, will be a major technique in the proof of the following transfer Theorems which we will give the statements without proofs. Cauchy's coefficients formula is a direct application of the known residue theorem in complex analysis.

We continue by defining some types of functions for which there exists some general theorems that allows mechanical procedures to extract asymptotic equivalent from them. The interesting part for us, is to understand to which type does the *generating function* of a combinatorial class belongs to. The simplest types of functions are the rational and meromorphic functions.

Definition 2.4.2. *A function $f(z)$ is a rational function, if and only if it is of the form $\frac{N(z)}{D(z)}$, where $N(z)$ and $D(z)$ are polynomials. Rational functions that are analytic at the origin, which is the case for some generating functions $D(0) \neq 0$.*

Definition 2.4.3. *A function $h(z)$ is meromorphic at z_0 if and only if, for z in a neighbourhood of z_0 with $z \neq z_0$, it can be represented as $\frac{f(z)}{g(z)}$, with $f(z)$ and $g(z)$ being analytic at z_0. In that case, it admits near z_0 an expansion of the form*

$$h(z) = \sum_{n \geq -M} h_n \, (z - z_0)^n.$$

If $h_{-M} \neq 0$ and $M \geq 1$, then $h(z)$ has a pole of order M at $z = z_0$ and the coefficient h_{-1} is called the residue of $h(z)$ at $z = z_0$.

The main goal of analytic methods is to derive asymptotic information from generating functions.

In simple words, a *singularity* of a function $f(z)$ is a point where $f(z)$ stops of being analytic. The search for the *dominant singularity* (the one that is nearest to the origin) of a generating is essential in deriving asymptotic information from a generating function.

Pringsheim's theorem applies to generating functions and allows one to restrict its attention to positive real axis in search for the dominant singularity of a combinatorial generating function (the power series has non-negative coefficients).

Theorem 2.4.4. *(Pringsheim's Theorem). If $f(z)$ is representable at the origin by a series expansion that has non-negative coefficients and radius of convergence R, then the point $z = R$ is a singularity of $f(z)$.*

The following result is known as The **First Principle of Coefficient Asymptotics** in Analytic combinatorics. The location of a function's singularities dictates the *exponential growth* (A^n) of its coefficient.

Theorem 2.4.5. *(Exponential Growth Formula). If $f(z)$ is analytic at 0 and R is the modulus of a singularity nearest to the origin. Then,*

$$f_n = \left(\frac{1}{R}\right)^n \theta(n),$$

where $\theta(n)$ is a subexponential factor, that is $\limsup |\theta(n)|^{\frac{1}{n}} = 1$.

The **Second Principle of Coefficient Asymptotics** relates *subexponential factors* of coefficients to the nature of singularities. For rational and meromorphic functions the results can be obtained simply.

For rational functions, the next theorem gives an exact finite expression for the coefficients of a function in term of its poles.

Theorem 2.4.6. *(Expansion of rational functions). If $f(z)$ is a rational function that is analytic at zero and has poles at points $\alpha_1, \alpha_2, \ldots, \alpha_m$, then its coefficients are a sum of exponential–polynomials: there exist m polynomials $\{\Pi_j(x)\}_{j=1}^{m}$ such that, for n larger than some fixed n_0,*

$$f_n = [z^n]f(z) = \sum_{j=1}^{m} \Pi_j(n)\alpha_j^{-n}.$$

The degrees of Π_j is equal to the order of the pole of $f(z)$ at α_j minus one.

We can define a similar expansion for meromorphic functions.

Theorem 2.4.7. *(Expansion of meromorphic functions). Let $f(z)$ be a function meromorphic at all points of the closed disc $|z| \leq R$, with poles at points $\alpha_1, \alpha_2, \ldots, \alpha_m$. Assume that $f(z)$ is analytic at all points of $|z| = R$ and at $z = 0$. Then there exist m polynomials $\{\Pi_j(x)\}_{j=1}^m$ such that:*

$$f_n = [z^n]f(z) = \sum_{j=1}^{m} \Pi_j(n)\alpha_j^{-n} + O\left(R^{-n}\right).$$

Furthermore the degree of Π_j is equal to the order of the pole of f at α_j minus one.

We give below an example of application of this Theorem that covers many different interesting combinatorial classes.

Example 2.4.8. *Let $f(z) = \dfrac{1}{2 - \exp(z)}$, which is the generating function of Surjections (also known as Ordered Bell numbers) as will be seen in Section 3.2. The function has a pole of order 1 at $\log 2$ which is the singularity of smallest modulus. The next pole is equal to $\log 2 + 2i\pi \approx 6.32130292$. From Theorem 2.4.7:*

$$f_n \underset{n\to\infty}{\sim} \Pi_1 \left(\frac{1}{\log 2}\right)^n,$$

where Π_1 is a polynomial of degree 0 (i.e a constant). To determine it we see that

$$f(z) \underset{z\to\log 2}{\sim} -\frac{1}{2}\frac{1}{z - \log 2},$$

So that asymptotically,

$$f_n \underset{n\to\infty}{\sim} \frac{1}{2}\left(\frac{1}{\log 2}\right)^{n+1}.$$

Some other very good accounts on methods for obtaining asymptotics of generating functions include researches of Bender, De Bruijn, Odlyzko and Wilf. See [**WdB60**, **Wil05**, **Odl95**, **Ben74**].

2.4.1 Singularity analysis

The main idea of the process of singularity analysis is the existence of a *correspondence between the asymptotic expansion of a function near its dominant singularities and the asymptotic expansion of the function's coefficients*.

It extends the analysis of meromorphic functions since it allows for functions whose singular expansion involves fractional powers and logarithms.

It is based on two ingredients : A *catalogue* of asymptotic expansions for coefficients of functions that are in standard scale (these functions occur in the singular expansion) and secondly, *transfer Theorems* which allow the extraction of the asymptotic order of the coefficients that are error terms in the singular expansion.

Its development is due to a pioneering paper of Flajolet and Odlyzko in 1990 [FO90] and is also detailed with examples in [FS09, Chapter VI].

Let $\mathcal{S}$ denote the set of the following singular functions:

$$\mathcal{S} = \{(1-z)^{-\alpha}\,\lambda(z)^{\beta} \mid \alpha, \beta \in \mathbb{C}\}, \qquad \lambda(z) = \frac{1}{z}\,\log\frac{1}{1-z} \equiv \frac{1}{z}L(z).$$

Definition 2.4.9. *(Delta domain) Given two numbers ϕ and R, with $R > 1$ and $0 < \phi < \frac{\pi}{2}$, the open domain $\Delta(\phi, R)$ is defined as*

$$\Delta(\phi, R) = \{z \mid |z| < R, z \neq 1, |arg(z-1)| > \phi\}.$$

A domain is a Δ-domain at 1 if it is a $\Delta(\phi, R)$ for some R and ϕ. For a complex number $\zeta \neq 0$, a Δ-domain at ζ is the image by the mapping $z \mapsto \zeta z\, z$ of a Δ-domain at 1. A function is Δ–analytic if it is analytic in some Δ-domain.

The *catalogue* of asymptotic expansions is based on the following two Theorems.

Theorem 2.4.10. *(Standard function scale). Let α be an arbitrary complex number in $\mathbb{C}\backslash\mathbb{Z}_{\leq 0}$. The coefficient of z^n in*

$$f(z) = (1-z)^{-\alpha}$$

admits for large n a complete asymptotic expansion in descending powers of n,

$$[z^n]f(z) = \frac{n^{\alpha-1}}{\Gamma(\alpha)}\left(1 + \sum_{k=1}^{\infty}\frac{e_k}{n^k}\right),$$

where e_k is a polynomial of degree $2k$. In particular:

$$[z^n]f(z) = \frac{n^{\alpha-1}}{\Gamma(\alpha)}\left(1 + \frac{\alpha(\alpha-1)}{2\,n} + \frac{\alpha(\alpha-1)(\alpha-2)(\alpha-3)}{24\,n^2} + O\left(\frac{1}{n^3}\right)\right).$$

Function	Coefficients
$(1-z)^{\frac{3}{2}}$	$\frac{1}{\sqrt{\pi\, n^5}}\left(\frac{3}{4} + \frac{45}{32\,n} + O\left(\frac{1}{n^2}\right)\right)$
$(1-z)^{\frac{1}{2}}\,\log\frac{1}{1-z}$	$-\frac{1}{\sqrt{\pi\, n^3}}\left(\frac{1}{2}\log n + \frac{\gamma + 2\log 2 - 2}{2} + O\left(\frac{\log n}{n}\right)\right)$
$(1-z)^{-\frac{1}{2}}\,\log\frac{1}{1-z}$	$\frac{1}{\sqrt{\pi\, n}}\left(\log n + \gamma + 2\log 2 + O\left(\frac{\log n}{n}\right)\right)$

Table 2.4: Some functions of the standard scale and the asymptotics of their coefficients according to Theorem 2.4.10 and Theorem 2.4.11

Theorem 2.4.11. *(Standard function scale, logarithms). Let α be an arbitrary complex number in $\mathbb{C}\backslash\mathbb{Z}_{\leq 0}$. The coefficient of z^n in the function*

$$f(z) = (1-z)^{-\alpha}\left(\frac{1}{z}\log\frac{1}{1-z}\right)^{\beta}$$

admits for large n a full asymptotic expansion in descending powers of $\log n$*,*

$$f_n = [z^n]f(z) = \frac{n^{\alpha-1}}{\Gamma(\alpha)}(\log n)^{\beta}\left(1 + \frac{C_1}{\log n} + \frac{C_2}{\log^2 n} + O\left(\frac{1}{\log^3 n}\right)\right),$$

where $C_k = \binom{\beta}{k}\Gamma(\alpha)\,\dfrac{d^k}{d\,s^k}\dfrac{1}{\Gamma(s)}|_{s=\alpha}$.

A summary of some applications of these two Theorems can be found in Table 2.4. They serve as a first basis for the process of singularity analysis. Some special cases for values of α and β are discussed in [**FS09**, p. 386].

The second part needed for the process of *singularity analysis* is the transfer Theorems. In the following we give the statement of the $Big - Oh$ transfer, however other transfer theorems exist.

Theorem 2.4.12. *(Transfer, Big-Oh). Let* α *and* β *be arbitrary real numbers, and let* $f(z)$ *be a function that is* Δ*–analytic. Assume that* $f(z)$ *satisfies in the intersection of a neighbourhood of* 1 *with its* Δ*–domain the condition*

$$f(z) = O\left((1-z)^{-\alpha}\,(\log\frac{1}{1-z})\right)^{\beta}.$$

Then,

$$[z^n]f(z) = O\left(n^{\alpha-1}\,(\log n)^{\beta}\right).$$

Staring from the expansion of a function at its singularity (its *singular expansion*) it is possible to justify term-by-term transfer which is the core of singularity analysis and by applying the *Standard scale* Theorems jointly with the *transfer* Theorems we get the following.

Theorem 2.4.13. *From* [**FS09**] *(**Singularity analysis**, single singularity). Let* $f(z)$ *be a function analytic at* 0 *with a singularity at* ζ*, such that* $f(z)$ *can be continued to a domain of the form* $\zeta\,.\,\Delta_0$*, for a* Δ*-domain* Δ_0 *where* $\zeta\,.\,\Delta_0$ *is the image of* Δ_0 *by the mapping* $z \mapsto \zeta z$*. Assume that there exist two functions* σ *and* τ*, where* σ *is a (finite) linear combination of functions in* $\mathcal{S}$ *and* $\tau \in \mathcal{S}$*, so that*

$$f(z) = \sigma(z/\zeta) + O(\tau(z/\zeta)) \qquad \textit{as} \quad z \to \zeta \quad \textit{in} \quad \zeta\,.\,\Delta_0.$$

Then, the coefficients of $f(z)$ *satisfy the asymptotic estimate*

$$f_n = \zeta^{-n}\sigma_n + O(\zeta^{-n}\tau_n^{\star}),$$

where $\sigma_n = [z^n]\sigma(z)$ *has its coefficients determined by Theorem 2.4.10 and Theorem 2.4.11 and* $\tau_n^{\star} = n^{a-1}\,(\log n)^b$*, if* $\tau(z) = (1-z)^{-a}\,\lambda(z)^b$.

As an illustration of this scheme we take the following example.

Example 2.4.14. *Let* $f(z) = \frac{\mathrm{e}^{-\frac{z}{2}-\frac{z^2}{4}}}{\sqrt{1-z}}$*. The nominator is entire while the denominator is singular at* 1*. However, the denominator is* Δ*-analytic and so does* $R(z)$*. We can write the singular expansion of* $R(z)$ *at* $z = 1$*. The factor* $(1-z)^{-\frac{1}{2}}$ *does not change so we expand* *the numerator and get,*

$$R(z) = \frac{\mathrm{e}^{-\frac{3}{4}}}{\sqrt{1-z}} + \mathrm{e}^{-\frac{3}{4}}\sqrt{1-z} + O\left((1-z)^{\frac{3}{2}}\right).$$

We can now use transfer Theorems for each term independently.

$$[z^n]\frac{\mathrm{e}^{-\frac{3}{4}}}{\sqrt{1-z}} = \frac{\mathrm{e}^{-\frac{3}{4}}}{\sqrt{\pi\, n}}\left(1 - \frac{1}{8n} + O\left(\frac{1}{n^2}\right)\right)$$

$$[z^n]\mathrm{e}^{-\frac{3}{4}}\sqrt{1-z} = \frac{-\mathrm{e}^{-\frac{3}{4}}}{2\sqrt{\pi\, n^3}}\left(1 + \frac{3}{8n} + O\left(\frac{1}{n^2}\right)\right)$$

As a conclusion we have,

$$[z^n]R(z) = \frac{\mathrm{e}^{-\frac{3}{4}}}{\sqrt{\pi\, n}} - \frac{5}{8}\frac{-\mathrm{e}^{-\frac{3}{4}}}{\sqrt{\pi\, n^3}} + O\left(\frac{1}{n^{\frac{5}{2}}}\right).$$

2.4.2 Asymptotics of linear differential equations

As it has been noted in Section 2.2 some operations on generating functions such as pointing give rise to differential operators. Especially, when the specification is recursive, it often leads to linear differential equations that may not be solvable with exact functions. However, if we are looking to extract asymptotic information the task does not require an exact solution and in many cases the asymptotics can be extracted mechanically from the differential equation.

The study of solutions of linear differential equations in the complex plane and their asymptotic expansions can be found in Wasow [**Was87**], Henrici [**Hen91**] and [**Inc44**]. Flajolet and Sedgewick in [**FS09**] reformulated some theorems for the Analytic combinatorics framework which we state some results.

Suppose that we have a *linear differential equation* (ODE) of the form

$$c_0(z)\,\partial^r y(z) + c_1(z)\,\partial^{r-1} y(z) + \cdots + c_r(z)\,y(z) = 0. \tag{2.1}$$

The integer r is said to be the order of the ODE, and it is assumed that there exists a simply connected domain Ω in which the coefficients $c_i(z)$ are analytic at a point z_0 where $c_0(z_0) \neq 0$. Therefore, in the neighbourhood of z_0 there exist r linearly independent analytic solutions of Equation (2.1). This is guaranteed by an existence theorem to be found in [**Was87**]. As a result singularities can only occur at points ζ that are roots of the leading coefficient $c_0(z)$.

For simplicity we will denote $c_j \equiv c_j(z)$ and rewrite Equation (2.1),

$$\partial^r y(z) + d_1(z)\,\partial^{r-1} y(z) + \cdots + d_r(z)\,y(z) = 0. \tag{2.2}$$

Where $d_i(z) = \dfrac{c_i(z)}{c_0(z)}$. As a result, the functions $d_i(z)$ are now meromorphic in Ω.

Let $f(z)$ be a meromorphic function. We define $\omega_\zeta(f)$ to be the **order** of the pole of f at ζ. In the following we give two definitions and the main asymptotic result.

Definition 2.4.15. *The differential equation in Equation (2.2) have a singularity at ζ if at least one of the $\omega_\zeta(d_j)$ is a positive. Moreover the point ζ is called a* ***regular singularity*** *if*

$$\omega_\zeta(d_1) \leq 1, \quad \omega_\zeta(d_2) \leq 2, \quad \ldots, \quad \omega_\zeta(d_r) \leq r,$$

The singularity is otherwise irregular.

In fact the case of *regular singularities* is more easy to handle and fortunately in this thesis all differential equations appearing will have regular singularities.

Definition 2.4.16. *Given an equation of the form of Equation (2.2) and a regular singular point ζ, the* ***indicial polynomial*** *$I(\theta)$ at ζ is defined as,*[3]

$$I(\theta) = \theta^{\underline{r}} + \delta_1 \theta^{\underline{r-1}} + \cdots + \delta_r,$$

where $\delta_j = \lim_{z \to \zeta} (z-\zeta)^j d_j(z)$.

The *indicial polynomial* is used to extract information about the dominant asymptotic behaviour. More formally, at a regular singular point, the Equation (2.2) transforms to

$$D[(z-\zeta)^\theta] = I(\theta)(z-\zeta)^{\theta - r} + O\left((z-\zeta)^{\theta-r-1}\right),$$

where D is the differential operator of Equation (2.2). The next Theorem shows the general form of solutions of these differential equations near a regular singularity ζ.

Theorem 2.4.17. *(Regular singularities of ODEs). Consider a meromorphic differential equation of the form Equation (2.2) and a regular singular point ζ. Assume that the indicial equation at ζ , $I(\theta) = 0$, is such that no two roots differ by an integer (in particular, all roots are distinct). Then, in a slit neighbourhood of ζ , there exists a linear basis of all the solutions that is comprised of functions of the form*

$$(z-\zeta)^{\theta_j} H_j(z-\zeta),$$

where $\theta_1, \theta_2, \ldots, \theta_r$ are the roots of the indicial polynomial and each H_j is analytic at 0. In the case of roots differing by an integer (or multiple roots), the solutions may include additional logarithmic terms involving non-negative powers of $\log(z-\zeta)$.

A specialisation of this last Theorem to account for cases where some log-terms might appear is given in the following which is a specialisation of Theorems found in [**Inc44**] and written in [**GGKW20**].

Theorem 2.4.18. *Consider a differential equation of the form Equation (2.2) and a regular singular point ζ such that $\omega_\zeta(d_i) = 1$ for all $i = 1, \ldots, r$, and $\delta_1 := \lim\limits_{z \to \zeta}(z-\zeta)d_1(z) \geq 0$. Then, the vector space of all analytic solutions defined in a slit neighbourhood of ζ has a basis of r functions, where $r-1$ functions are of the form*

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

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

[3] $n^{\underline{k}} = n(n-1)\ldots(n-k+1)$ represents the descending factorial.

(1) For $\delta_1 \in \{0, 1, \dots, r-1\}$ it is of the form

$$(z-\zeta)^{r-1-\delta_1}\, H(z-\zeta)\, \log(z-\zeta),$$

(2) For $\delta_1 \in \{r, r+1, \dots\}$ it is of the form

$$(z-\zeta)^{r-1-\delta_1}\, H(z-\zeta) \,+ H_0(z-\zeta)\,(\log(z-\zeta))^k, \quad \textit{with} \quad k \in \{0,1\},$$

(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 $H(0) \neq 0$.

For linear differential equations the constants in the asymptotic development are usually hard to compute. So that our results will depend in general on some constant that exists but for which we do not know the value.

As an example of application of this last theorem we give the following.

Example 2.4.19. *Let $h(z)$ be analytic at 0 and satisfy*

$$(-2\,z-1)\,\partial\, h\,(z) + (1-z)\,\partial^2\, h\,(z) = 0,$$

with initial conditions $h(0) = 0$ and $h'(0) = 1$. The equation is in the form of Equation (2.1), after dividing by the highest derivative we get,

$$\partial^2\, h\,(z) + \frac{(-2\,z-1)\,\partial\, h\,(z)}{1-z} = 0.$$

The value $z = 1$ is a regular singularity and moreover $\omega_1(\dfrac{(-2\,z-1)}{1-z}) = 1$. We have,

$$\delta_1 = \lim_{z\to 1} \frac{(-2\,z-1)\,(z-1)}{1-z} = 3.$$

Then the basis of solutions around $z = 1$ contains 2 functions: $H_0(z-1)(z-1)$ and $H_2(z-1)(z-1)^{-2} + H_3(z-1)(\log(z-1))^k$ with $k \in \{0,1\}$ where $H_i(z-1)$ are analytic functions around $z = 1$. Finally we find,

$$[z^n]h(z) \underset{n\to\infty}{\sim} c\,n,$$

for some constant c that depends on the value of the function $H_2(0)$.

CHAPTER 3

Classical objects in Combinatorics

Contents

Die ganzen Zahlen hat der liebe Gott gemacht, alles andere ist Menschenwerk.[1]

Leopold Kronecker (1823 - 1891), Quoted by Weber in Jahresbericht der Deutschen Mathematiker-Vereinigung 1891-92.

[1] God made the integers, all else is the work of man

This chapter is dedicated to the presentation of the main classical combinatorial classes and objects that we will be using frequently as well as the notations that we will adopt for them.

Combinatorics is an area of mathematics concerned with counting and studying certain properties of finite structures. It is closely related to many other areas of mathematics and has many applications ranging from logic to statistical physics, from evolutionary biology to computer science, etc.

The first question that arise after defining a combinatorial class of objects is usually the enumeration problem, in which ideally we look for an explicit formula, or at least finding some recurrence relation that counts the objects.

An important question arises when two sets of objects have the same cardinality (same number of objects) then we usually seek to find bijections between these two sets.

Then comes the question of statistics and random generating. What is the typical shape of an object taken randomly in the set of all possible objects. A related question is the question of random generation. Is it possible to efficiently generate random objects in the set of all possible ones?

Counting objects to our knowledge dates back to earliest civilisations such as Babylonians, Indians and Greeks. In the 6th century BCE, ancient Indian physician Sushruta asserts in Sushruta Samhita that 63 combinations can be made out of 6 different tastes, taken one at a time, two at a time, etc., thus computing all $2^6 - 1$ possibilities. An interesting account on the roots of combinatorics have been published by Biggs in [**Big79**].

A famous example is also given by a page of Plutarch's Moralia where following statement appears "Chrysippus says that the number of compound propositions that can be made from only ten simple propositions exceeds a million. (Hipparchus, to be sure, refuted this by showing that on the affirmative side there are 103049 compound statements, and on the negative side 310 952.)"

This was related by Stanley in [**Sta97**]. The number 103049 corresponds exactly to the 10-th number of what is nowadays known as Hipparchus of Rhodes and Schröder numbers referenced under EIS A001003[2]. We will come back to these numbers in Section 3.5.3 to define them more precisely and see their relationship to tree structures as suggested by Schröder.

3.1 Permutations

Permutations are one of the most famous object of study in combinatorics. The permutations of a set of n elements represent the number of possible ways to rearrange the elements between each others. In a paper [**Bro11**] of 2011 by Broemeling talks of an Arab mathematician and cryptographer Al-Khalil who lived in the 8-th century, , wrote the Book of Cryptographic Messages. It contains the first known use of permutations and combinations, to list all possible Arabic words with and without vowels.

[2] Throughout this thesis, a reference EIS A$\cdots$ points to Sloane's Online Encyclopedia of Integer Sequences `www.oeis.org`.[**Slo06**]

The way of determining the number of permutations of a set of n elements was also known to Indians. Bhāskara an Indian mathematician and astronomer who lived in the 12-th century wrote a treatise in mathematics in which we find: ”The product of multiplication of the arithmetical series beginning and increasing by unity and continued to the number of places, will be the variations of number with specific figures.”

This was reported by by Biggs in [**Big79**]. He also cites a second example that involves references to permutations which the medical treatise of Susruta, which may be as old as the 6th century B.C. However, it is difficult to date this document with certainty.

Permutations can be defined as bijections from a set P onto itself. All permutations of a set with n elements form a symmetric group, denoted $\mathbf{P}_n$,where the group operation is the functional composition. Then we have that,

$$|\mathbf{P}_n| = n! = n \cdot (n-1) \cdot \cdots \cdot 2 \cdot 1$$

A permutation of n elements can be represented in a two line matrix where each element of the first line is sent to its corresponding place in the rearrangement see Figure 3.1 for an example.

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 6 & 2 & 5 & 1 \end{pmatrix}$$

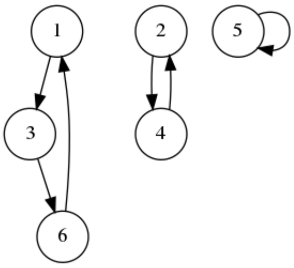


Figure 3.1: A permutation of 6 elements and its corresponding graphical representation.

The first line can be omitted if the context is not ambiguous. Throughout this thesis we will denote a single permutation by usually by $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ and call it the standard notation. We will also introduce the cycle notation in the next section. So the permutation in Figure 3.1 can be written simply as $\sigma = (3, 4, 6, 2, 5, 1)$ and $\mathcal{S}_n$ for the set of all permutations of n elements.

The ***orbit*** of an element x under the action of a permutation σ denoted by $\mathcal{O}_\sigma(x)$ is the subset of successive applications of σ on x that is,

$$\mathcal{O}_\sigma(x) = \{\sigma^k(x), k \in \mathbb{N}\}.$$

Permutations can be used in many different areas such that computer science, quantum physics, and biology to model RNA sequences. Some of the most important books in the literature on permutations include [**Bón12, GKP94, FS09, Knu05**].

There are many statistics that can be defined on permutations and extensive ongoing researches are dedicated to them. We give in the next sections some statistics that we will be use.

As for their specification in the language of symbolic method presented in Section 2.2. A permutation is a sequence of integers. In the labelled universe this gives:

$$\mathcal{P} = \mathrm{Seq}\left(\mathcal{Z}\right),$$

that translates to,

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

The coefficients of this sequence are found in EIS A000142. Alternatively we can use Remark 2.2.5 and get that $\mathcal{P}$ can also be specified as a set of cycles. In the next section we see more details about this last decomposition.

3.1.1 Cycles

A ***cycle*** in a permutation σ, correspond to a subset of elements of that permutation whose elements trade places with one another. On the group level a cycle in a permutation σ of n elements corresponds to the *orbit* of one element of this permutation. Finally, the notion of cycle also corresponds to the cycles that appear in the graphical representation of the permutation.

Two different elements in a permutation can be in the same cycle. A permutation can always be decomposed uniquely into a set of disjoint cycles. For example the permutation in Figure 3.1 can be seen as $\sigma = (3,4,6,2,5,1) = (1,3,6)(2,4)(5)$.

Therefore, we can write another specification for permutations in the language of symbolic method based on Remark 2.2.5. A permutation is simply a set of disjoint cycles of numbers.

$$\mathcal{P} = \mathrm{Set}(\mathrm{Cyc}(Z)),$$

which gives

$$P(z) = \exp\left(\log\frac{1}{1-z}\right) = \frac{1}{1-z}.$$

Now if we take the set of all permutations of size n denoted $\mathcal{S}_n$ and partition its element following the number of cycles each permutation has. A permutation can have between 1 and n cycles. This partitioning is well known and the numbers are called ***Stirling cycle numbers*** (or *Stirling numbers of the first kind*) that we will denote by $\left[{n \atop k}\right]$. See Table 3.1 for the first values of Stirling cycle numbers. These numbers are referenced in EIS A132393.

1						
1	1					
2	3	1				
6	11	6	1			
24	50	35	10	1		
120	274	225	85	15	1	
720	1764	1624	735	175	21	1

Table 3.1: Stirling cycle numbers for $n \in \{1, \dots, 7\}$ and $k \in \{1, \dots, n\}$

A recurrence relation for these numbers is defined as follows. $\left[{0 \atop 0}\right] = 1$. Then $\forall n > 0$, $\left[{n \atop 0}\right] = 0$ and $\left[{n \atop n}\right] = 1$. Then when $n \geq 2$ and $1 \leq k \leq n-1$

$$\begin{bmatrix} n \\ k \end{bmatrix} = \begin{bmatrix} n-1 \\ k-1 \end{bmatrix} + (n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix}.$$

The ***signature*** of a permutation σ of size n denoted $sgn(\sigma)$ is defined as

$$sgn(\sigma) = (-1)^{n-m},$$

Where m is the number of cycles of σ. We say that the permutation is **even** if $sig(\sigma) = 1$ and **odd** if it is -1.

Now, we can define the ***alternating group*** (also **group of even permutation**) which is denoted Alt_n to be the subgroup of $\mathbf{P}_n$ that contains even permutations.

3.1.2 Eulerian numbers and runs

We give two successive definitions of descent and runs and see the relation between each others.

Definition 3.1.1. *From* [**Bón12**] *Let σ be a permutation of $\mathcal{S}_n$ then i is a* ***descent*** *of σ, if $\sigma_i > \sigma_{i+1}$. Similarly, we say that i is an* ***ascent*** *or* ***rise*** *if $\sigma_i < \sigma_{i+1}$.*

For example, $\sigma = (3,4,6,2,5,1)$, has 2 descents which are $\{6,2\}$ and $\{5,1\}$. A set of ascending sequences in a permutation is called a ***run*** in [**GKP94**] or a *rise* in [**Com12**]. Formally we give the following definition.

Definition 3.1.2. *An ascending run (respectively a descending run) of a permutation σ is a maximal increasing (respectively decreasing) sub-sequence. That is $(\sigma_i, \sigma_{i+1}, \ldots, \sigma_j)$ $(1 \leq i \leq j \leq n)$ such that if $i \leq p \leq q \leq j$, then $\sigma_p \leq \sigma_q$ (respectively $\sigma_p \geq \sigma_q$).*

For example, the ascending runs of $\sigma = (3,4,6,2,5,1)$ are $\{3,4,6\}$, $\{2,5\}$ and $\{1\}$. The number of permutations of size n with k descents are called Eulerian numbers which are denoted $\left\langle {n \atop k} \right\rangle$. We also denote by $r(n,k)$ the number of permutations of size n having k ascending runs. We have the following result. If σ has $k-1$ descents, then σ can be decomposed into k as-ending runs. Which leads to the following equality

Result 3.1.3. $\forall n \geq 1, k \geq 1,$

$$\left\langle {n \atop k-1} \right\rangle = r(n,k).$$

There is a simple recurrence relation on these numbers. We have that for, $\left\langle {1 \atop 0} \right\rangle = 1$ and for $n \geq 2$, and $k = 0$ or $k = n-1$, $\left\langle {n \atop k} \right\rangle = 1$. Then for $n \geq 3, 1 \leq k \leq n-2$

$$\left\langle {n \atop k} \right\rangle = (n-k)\left\langle {n-1 \atop k-1} \right\rangle + (k+1)\left\langle {n-1 \atop k} \right\rangle.$$

The first values of the Eulerian numbers are depicted in Table 3.2 and the table can be find in EIS A008292. These numbers have remarkable properties there is a wealth of literature about them. Some accounts are [**Knu98, Cha08, FS06, Car59**].

1						
1	1					
1	4	1				
1	11	11	1			
1	26	66	26	1		
1	57	302	302	57	1	
1	120	1191	2416	1191	120	1

Table 3.2: Eulerian numbers for $n \in \{1, \ldots, 7\}$ and $k \in \{0, \ldots, n\}$

3.2 Set partitions and Surjections

A ***set partition*** is a partition of finite set into a number of non-empty subsets (also called boxes). For example there is 1 partition of 3 elements into 1 subset, 3 into 2 subsets and 1 into 3 subsets as depicted in Figure 3.2.

These numbers have been studied in the 19th century even though they were known from more ancient times. See [**Con12, Gar78, Knu13, FS09**] for some references on the subject.

The total number of *set partitions* for a fixed size n is counted by what is called ***Bell numbers*** which we will denote b_n. We already saw that $b_3 = 5$.

$$(S_n)_{n \geq 0} = 1, 1, 2, 5, 15, 52, 203, 877, 4140, 21147, 115975, \ldots$$

It is referenced under EIS A000110.

1 subset	2 subsets	3 subsets
$\{\{1,2,3\}\}$	$\{\{1,2\},\{3\}\}$	$\{\{1\},\{2\},\{3\}\}$
	$\{\{1,3\},\{2\}\}$	
	$\{\{2,3\},\{1\}\}$	

Figure 3.2: The 5 partitions of the set $\{1, 2, 3\}$.

Let $\mathcal{S}$ be the class of set partitions. Using the symbolic method we see that set partitions are sets of non-empty sets of integers, therefore the **EGF** of $\mathcal{S}$,

$$\mathcal{S} = \text{Set}\left(\text{Set}_{\geq 1}\left(\mathcal{Z}\right)\right).$$

Which gives

$$S(z) = \mathrm{e}^{\mathrm{e}^{z}-1}.$$

Let us now define the numbers $\left\{ {n \atop k} \right\}$ which represents the number of set partitions of a set n elements into k non-empty subsets. These numbers are called ***Stirling partition numbers*** (or *Stirling numbers of the second kind*). See Table 3.3 for the first values of $\left\{ {n \atop k} \right\}$.

It is clear that S_n can be defined as a sum of $\left\{ {n \atop k} \right\}$,

$$S_n = \sum_{k=1}^{n} \left\{ {n \atop k} \right\}.$$

1						
1	1					
1	3	1				
1	7	6	1			
1	15	25	10	1		
1	31	90	65	15	1	
1	63	301	350	140	21	1

Table 3.3: Stirling partition numbers for $n \in \{1, \ldots, 7\}$ and $k \in \{1, \ldots, n\}$

Once again, Stirling partition numbers have a simple recurrence relation. $\left\{ {0 \atop 0} \right\} = 1$. For $n > 0$, $\left\{ {n \atop 0} \right\} = 0$ and $\left\{ {n \atop n} \right\} = 1$. Then for $n \geq 2$, and $1 \leq k \leq n-1$

$$\left\{ {n \atop k} \right\} = \left\{ {n-1 \atop k-1} \right\} + k \left\{ {n-1 \atop k} \right\}.$$

Now, if we put an ordering on the boxes that contain the partition of the set we get what is called an ***Ordered set partition*** or ***Surjections***. The number of ordered set partitions of fixed size n is known as ***Ordered Bell numbers*** (also called *Fubini numbers*).

$$(B_n)_{n \geq 0} = 1, 1, 3, 13, 75, 541, 4683, 47293, 545835, 7087261, 102247563, \ldots$$

It can be found in EIS A000670. The ordered partitions of a set of 3 elements are depicted in Figure 3.3

1 subset	2 subsets	3 subsets
$[\{1,2,3\}]$	$[\{1,2\},\{3\}]$	$[\{1\},\{2\},\{3\}]$
	$[\{3\},\{1,2\}]$	$[\{2\},\{1\},\{3\}]$
	$[\{1,3\},\{2\}]$	$[\{1\},\{3\},\{2\}]$
	$[\{2\},\{1,3\}]$	$[\{2\},\{3\},\{1\}]$
	$[\{2,3\},\{1\}]$	$[\{3\},\{1\},\{2\}]$
	$[\{1\},\{2,3\}]$	$[\{3\},\{2\},\{1\}]$

Figure 3.3: The 13 Ordered partitions of the set $\{1, 2, 3\}$.

It is still possible to write a sum for B_n in term of $\left\{ {n \atop k} \right\}$ by multiplying the numbers by $k!$ and call them *Ordered Stirling partition numbers*.

$$S_n = \sum_{k=1}^{n} k! \left\{ {n \atop k} \right\}.$$

Once again if we let $\mathcal{B}$ be the class of ordered partition, using the **EGF** scheme of the symbolic method we see that this time we have sequences non-empty set, thus,

$$\mathcal{B} = Seq\left(\mathrm{Set}_{\geq 1}\left(\mathcal{Z}\right)\right).$$

Which leads to

$$B(z) = \frac{1}{1 - (\mathrm{e}^z - 1)} = \frac{1}{2 - \mathrm{e}^z}.$$

1					
1	2				
1	6	6			
1	14	36	24		
1	30	150	240	120	
1	62	540	1560	1800	720
1	126	1806	8400	16800	15120

Table 3.4: Ordered Stirling partition numbers for $n \in \{1, \ldots, 7\}$ and $k \in \{1, \ldots, n\}$

3.3 Integer partitions and compositions

It is possible that prehistorical used astragalus of huckle-bone of certain animals to play games with them. Since a grouped number of these bones are not rare to be found during archaeological excavations. But ancient Egyptians used these bones to determine moves in simple board games. But it is during middle ages that systematic use of of dice games. The first studies on this subject goes back to Cardano (1501 - 1576) with a book titles “Book on game of chance” and a manuscript by Galileo (1564 - 1642) with some frequencies on sums of dices, see [**Big79**] for more details.

An ***integer partition*** is a way of writing an integer n as a sum of positive integers. If the order of the elements in the sum matters then it is called an ***integer composition***.

To illustrate the differences between integers partitions and compositions see Figure 3.4 and Figure 3.5.

Compositions can be seen as sequences of integers. It we denote $\mathcal{C}$ to be the class of *integer compositions*, then in the realm of **OGF** the class of positive integers can be written as $\mathcal{I} = \text{Seq}_{\geq 1}(Z)$. We can then write:

$$\mathcal{C} = \text{Seq}(\mathcal{I}),$$

which translates to,

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}.$$

By coefficient extraction wee see easily that,

$$C_n = 2^{n-1}.$$

The first coefficients are,

$$(C_n)_{n\geq 1} = 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, \ldots$$

This sequence is referenced under EIS A000079. Let us define $C_{n,k}$ to be the number of composition of integer n into k parts. We can write the integer n as a sequence of points and put $k-1$ bars to split this integer into k distinct parts. The number of possible places to place the bars is $n-1$ and we want to place $k-1$ of them. Therefore,

$$C_{n,k} = \binom{n-1}{k-1}.$$

$$\begin{array}{c} 4 \\ 3+1 \\ 2+2 \\ 2+1+1 \\ 1+1+1+1 \end{array}$$

Figure 3.4: The 5 integer partitions of 4.

Partitions on the other hand can be specified using multi-sets of elements. Since the ordering of the summands does not matter. If we let $\mathcal{IP}$ be the class of integer partitions then,

$$\mathcal{IP} = \text{MSet}\,(I)\,,$$

So that,

$$IP(z) = \prod_{m\geq 1} \frac{1}{(1-z^m)}.$$

The first coefficients are,

$$(IP_n)_{n\geq 1} = 1, 1, 2, 3, 5, 7, 11, 15, 22, 30, 42, 56, 77, 101, 135, \ldots$$

The sequence can be found in EIS A000041.

$$\begin{array}{c} 4 \\ 3+1 \\ 1+3 \\ 2+2 \\ 2+1+1 \\ 1+2+1 \\ 1+1+2 \\ 1+1+1+1 \end{array}$$

Figure 3.5: The 8 integer compositions of 4.

3.4 Graphs

A *graph* is a set of vertices together with links between pairs of vertices.

Definition 3.4.1. *A graph (or an undirected graph) is a pair $G = (V, E)$, where V is a set whose elements are called vertices, and $E \subseteq \{\{x, y\} \mid (x, y) \in V^2\}$ is a set of pairs (sets with two distinct elements) of vertices, whose elements are called edges.*

The degree of a vertex v in a graph G is denoted $D_G(v)$ or simply $D(v)$. It is defined by the number of vertices adjacent to v (also called the neighbors).

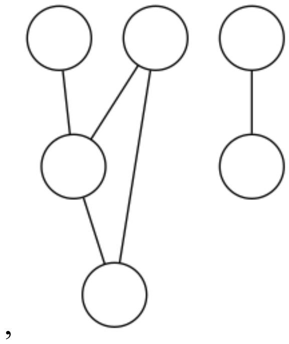


Figure 3.6: A graph of 6 vertices

If the order of links between vertices matters then we have a directed graph which has the same definition as graph except that the set of edges is not a two-set elements but pairs of elements in which $(x, y) \neq (y, x)$

Definition 3.4.2. *A directed graph is a pair $G = (V, E)$, where V is a set whose elements are called vertices, and $E \subseteq \{(x, y) \mid (x, y) \in V^2\}$ is a set of ordered pairs of vertices.*

In Figure 3.7, we put a graph and a directed graph We can also define a *multigraph* if there could be several links between the same pairs of elements.

Definition 3.4.3. *A labelled graph (respectively labelled directed graph) is a graph (respectively directed graph) in which each vertex has a fixed label.*

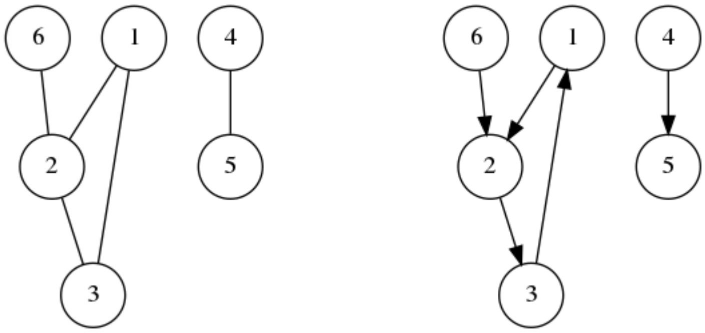


Figure 3.7: (left)A labelled graph and (right) a directed labelled graph both of 6 vertices

Definition 3.4.4. *A multigraph (respectively directed multigraph) is a pair $G = (V, E)$, where V is a set whose elements are called vertices, and E is a multiset of pairs of pairs (respectively ordered pairs) of vertices.*

As in the case of graphs, multigraphs can also be labelled to form *labelled multigraphs*.

Result 3.4.5. *Let G_n be the number of graphs on n labeled nodes then for $n \geq 1$,*

$$G_n = 2^{\binom{n}{2}}$$

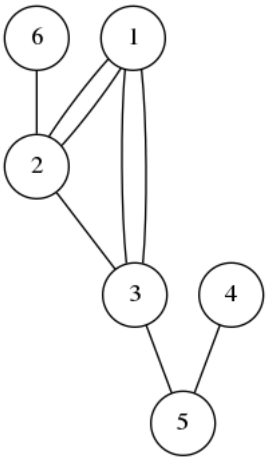


Figure 3.8: A labelled multigraph of 6 vertices

The result can be seen easily. Since the set of vertices has n elements. Then the set of edges has $E = \binom{n}{2}$ elements. Finally the number of possible graphs is the powerset of the set of edges which represent all combinations to take different edges. $G_n = 2^E$ The sequence G_n is referenced as EIS A006125.

A *connected graph* is a graph in which there is a path between any pair of vertices.

3.5 Trees

Trees are important structures, they are used extensively in computer science for compilers, sorting algorithms, efficient database, representing lists,.... Their study is also relevant in biology and phylogenetics and many other areas. In mathematics, people have been interested in the systematic study of their properties. Modern studies date back to Francis Galton and his investigation of the extinction of family names [**Gal73**] as well as Ernst Schröder in [**Sch70**]. Some good introductions to trees in combinatorics and probability theory can be found in [**Drm09**, **FS09**, **CCG18**].

Formally a *tree* is a connected graph without cycles.

Definition 3.5.1. *A rooted tree is a tree where a certain node is distinguished called the root node.*

Throughout this thesis we will be treating rooted trees and in the figures we will draw them upside-down such that the root node is always at the top, see Figure 3.9 for an example.

Trees are planar graphs since they can be embedded into a plane without crossings. However a tree can have different embeddings. Therefore counting families of trees with different embeddings or not makes a difference. If we are counting a family of planar trees then we need to consider all different embeddings of the tree.

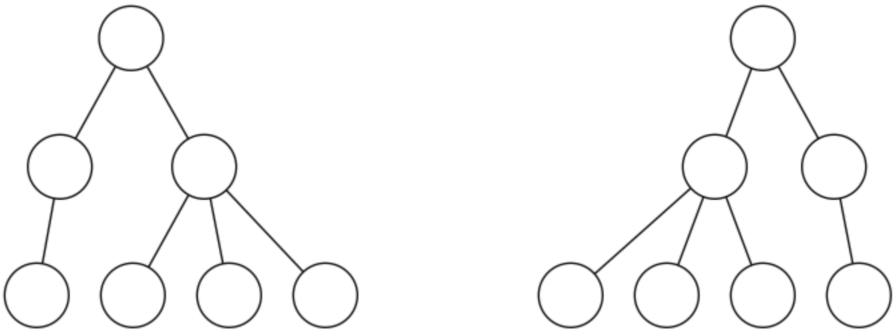


Figure 3.9: (left) An ordered rooted and (right) a different ordered rooted tree, both trees have 7 nodes.

The vocabulary of trees will be a bit different from the graph one. Therefore, we will refer to the vertices of the tree as *nodes*. We say that a tree r is a *subtree* of a if a is the parent of the root if r (we also say that r is a *child* of a). The *degree* (or *arity*) of a node a in t is its number of childen which corresponds to $D(a) - 1$ on the underlying graph structure (the minus 1 is to substract the parent node). This will be denoted by $d_t(a)$ or $d(a)$.

Finally, nodes of degree > 0 are called *internal nodes* while nodes of degree 0 are referred to as *leaves*.

Definition 3.5.2. *A planar tree (or ordered tree) is a tree where subtrees of a common node are ordered between each others (and represented from left to right).*

In Figure 3.9, if we are considering planar trees then the two trees are different. However the underlying graph is the same. Therefore if we were counting non-plane trees these two would be the same.

Definition 3.5.3. *A labelled tree is a tree which the underlying graph is a labelled graph.*

Remark 3.5.4. *Usually the term labelled trees is used to denote a tree where each node has a different label, and if the tree has n nodes then the labels range from 1 to n. But it is more suitable to call it a bijective labelling between the nodes and the set $[n]$.*

Definition 3.5.5. *A tree that has n nodes is a* ***weakly labelled tree*** *if each node is labelled by an integer, and if k is the maximum label in the tree, then all labels from 1 to k appear in the tree.*

It is easy to see that the number of different labellings (or bijective labellings) of a tree with n nodes is $n!$.

The number of weak labellings of a tree is counted by *ordrered Bell numbers* (also *ordrered set partitions*), see Section 3.2 for more details on it. We need to define an ordering on the nodes of the tree so that we can talk of first, second node, etc.

The idea is to take an ordered set partition of n. If the partition has k parts, then the tree will be labelled from 1 to k. The integers in the first subset of the partitions represent the nodes that take label 1 and so on.

3.5.1 Simple varieties of trees

Have been introduced by Meir and Moon in [MM78]. We start by defining the concept of a degree function which is a function that groups all possible arities that a *simple tree* can be built with.

Definition 3.5.6. *Let Ω be a multiset of integers that does not contain 0. Then the colored degree function $\phi(u)$ of Ω is:*

$$\phi(u) = \sum_{\omega \in \Omega} u^{\omega}.$$

Each different *colored degree function* gives a different variety of simple trees. The ***colored degree function*** defines the set of allowed arities and colors for the nodes in a tree. It is then possible to write $\phi(u)$ as a power series

$$\phi(u) = \sum_{n \geq 1} \phi_n u^n.$$

Therefore ϕ_i represents the number of colors of nodes of arity i.

Definition 3.5.7. *Let $\phi(u)$ be a colored degree function. The class $\mathcal{T}^{\phi}$ of **simple trees** or (simply generated trees) parametrized by $\phi(u)$ contains all trees that are rooted, plane and unlabelled. Each internal node has a degree d and a color c (d, c) such that the degree and the color belongs to Ω (i.e $[u^d]\phi(u) \geq 1$ and $c \leq [u^d]\phi(u)$). The size of a tree is given by its total number of nodes.*

Some examples of simple varieties of tree are given in the following.

Example 3.5.8. *Binary trees. Take $\Omega = \{1, 1, 2\}$ which gives $\phi(u) = 2u + u^2$. The number of binary trees of size n is known to be the famous Catalan numbers referenced under EIS A000108.*

Example 3.5.9. *Proper binary trees. Take $\Omega = \{2\}$ which gives $\phi(u) = u^2$. The number of proper binary trees of size n corresponds to shifted Catalan numbers that include 0 on even indices because there are no trees of even sizes. They are referenced under EIS A126120.*

Example 3.5.10. *Plane trees. Take $\Omega = \{1, 2, \dots\}$ which gives $\phi(u) = \frac{u}{1-u}$. This class of tree is also counted by Catalan numbers.*

Example 3.5.11. *Motzkin trees or (unary-binary trees). $\Omega = \{1, 2\}$ with $\phi(u) = u + u^2$. EIS A178834. See Figure 3.10 for an example.*

Example 3.5.12. *k-ary trees. $\Omega = \{k\}$ with $\phi(u) = u^k$.*

From Definition 3.5.7 of simple trees we can specify these families in the OGF universe. Let $\phi(u)$ be a degree function, then,

$$\mathcal{T} = \mathcal{Z} \times (1 + (\phi \circ \mathcal{T}))$$

In other word a tree is either a leaf or an internal node with a degree and a color as in ϕ. From the specification we get

$$T(z) = z\left(1 + (\phi(T(z))\right). \tag{3.1}$$

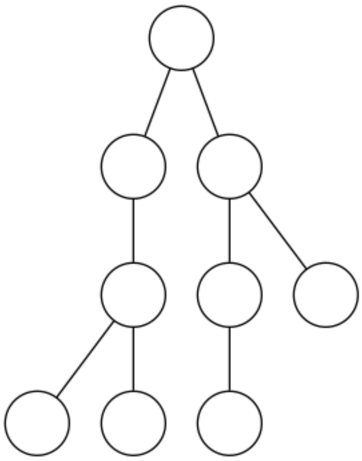


Figure 3.10: An example of a Motzkin tree

The asymptotic counting of varieties of simple trees shows a universal behavior with a typical polynomial of order $n^{-3/2}$. A complete study of this phenomena can be found in [**FS09**, p. 452]. We give here the statement of the main Theorem. But we need to start first with two technical conditions on the *colored degree function* $\phi(u)$. Let $\hat{\phi}(u) = 1 + \phi(u)$

Condition 3.5.13. *The function $\hat{\phi}(u)$ is such that*

$$\hat{\phi}(0) \neq 0, \quad [u^n]\hat{\phi}(u) \geq 0, \quad \hat{\phi}(u) \not\equiv \hat{\phi}_0 + \hat{\phi}_1 u.$$

Condition 3.5.14. *Within the open disk of convergence of $\hat{\phi}$ at $|z| < R$, there exists (necessarily unique) positive solution to the characteristic equation:*

$$\exists \tau, 0 \leq \tau \leq R, \quad \hat{\phi}(\tau) - \tau\hat{\phi}'(\tau) = 0.$$

A class of tree that satsifies these conditions is said to belong to the *smooth inverse-function* schema. The schema is said to be aperiodic if $\hat{\phi}(u)$ is an aperiodic function of u.

Theorem 3.5.15. [**FS09**, Theorem VII.2] *Let $y(z)$ belong to the smooth inverse-function schema (i.e it satsifies Condition 3.5.13 and Condition 3.5.14) in the aperiodic case. Then, let τ be the positive root of the characteristic equation and $\rho = \tau/\hat{\phi}(\tau)$, we have,*

$$[z^n]y(z) = \sqrt{\frac{\hat{\phi}(\tau)}{2\hat{\phi}''(\tau)}} \frac{\rho^{-n}}{\sqrt{\pi\, n^3}} \left(1 + O\left(\frac{1}{n}\right)\right).$$

3.5.2 Pólya trees

Pólya trees are rooted, non-plane trees where the size of the tree is taken to be its total number of nodes. Using the symbolic method seen in Section 2.2. We can define the class $\mathcal{H}$ of Pólya trees as follows:

$$\mathcal{H} = \mathcal{Z} \times \text{MSet}(\mathcal{H}).$$

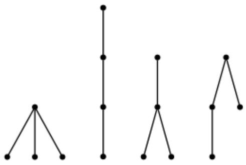


Figure 3.11: All 4 Pólya trees of size 4.

There is no known close formula for the resulting generating function but nontheless they can be studied for asymptotic enumeration using the functional equation resulting from the specification. The first values of H_n are:

$$0, 1, 1, 2, 4, 9, 20, 48, 115, 286, 719, 1842, 4766, 12486, \ldots$$

The sequence can be found in EIS A000081 and the four trees of size 4 are depicted in Figure 3.11.

3.5.3 Schröder trees

Schröder trees have first been studied by Ernst Schröder in a famous paper of 1870 [**Sch70**].

Definition 3.5.16 (see [**FS09**, p. 69])**.** *A Schröder tree is a rooted plane tree whose internal nodes all have arity at least* 2*. The size of a Schröder tree is its number of leaves.*

Note that a Schröder tree is an unlabelled combinatorial structure (neither the leaves nor the internal nodes are labelled). In the context of analytic combinatorics the combinatorial class $\mathcal{S}$ of Schröder trees is thus specified as

$$\mathcal{S} = \mathcal{Z} \cup \mathrm{Seq}_{\geq 2}\,\mathcal{S}. \tag{3.2}$$

Its combinatorial meaning is direct in the context of decomposable objects (see Flajolet and Sedgewick [**FS09**] for a detailed introduction to the combinatorial specification): An object from $\mathcal{S}$ is either a leaf (represented by the single atom $\mathcal{Z}$, of size 1), or it is composed of an internal node, parent of a sequence of at least two elements from $\mathcal{S}$. Not that, in the specification, the internal nodes are omitted (because they are of size 0): the expression

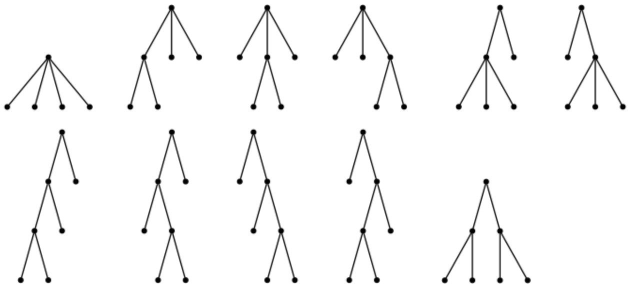


Figure 3.12: All 11 Schröder trees of size 4.

$\operatorname{Seq}_{\geq 2} \mathcal{S}$ is a abbreviation of $\mathcal{E} \times \operatorname{Seq}_{\geq 2} \mathcal{S}$ where $\mathcal{E}$ stands for an atom of size 0 and $\operatorname{Seq}_{\geq 2} \mathcal{S}$ is a sequence of at least two elements from $\mathcal{S}$.

Once the combinatorial specification is given, the classical *symbolic method* presented in Section 2.2, translates automatically the equation specifying the objects into a functional equation satisfied by the (ordinary) generating functions associated to the structures. The generating function of $\mathcal{S}$ is defined as the formal series $S(z) = \sum_{n\geq 1} s_n z^n$ where s_n is the number of Schröder trees of size n (i.e. with n leaves). Using the symbolic method on Equation (3.2), we get that

$$S(z) = z + \frac{S(z)^2}{1 - S(z)}. \tag{3.3}$$

An elementary iteration allows us to extract the first coefficients of the sequence $(s_n)_{n\in\mathbb{N}}$:

$$(0, 1, 1, 3, 11, 45, 197, 903, 4279, 20793, 103049, 518859, 2646723, 13648869, 71039373, \dots).$$

Equation (3.3) implies that the generating function S is algebraic and in fact

$$S(z) = \frac{1 + z - \sqrt{1 - 6z + z^2}}{4}.$$

This is sufficient to get the following asymptotic equivalent of s_n when n tends to infinity:

$$s_n = \frac{\sqrt{3\sqrt{2} - 4}}{4\sqrt{\pi}}\, n^{-3/2} \left(3 - 2\sqrt{2}\right)^{-n} \left(1 + \mathcal{O}(1/n)\right).$$

We refer the reader to [**FS09**, page 69] for a more detailed analysis of this generating function S. Figure 3.12 depicts all trees of size 4. Variants of Schröder trees can be defined, by allowing the node degrees in some fixed set. For instace:

$$\mathcal{S} = \mathcal{Z} \cup (\mathcal{S} \times \mathcal{S} \times \mathcal{S}). \tag{3.4}$$

Which gives Shröder such that internal nodes have arity 3.

Remark 3.5.17. *In we fix the node degree of a Schröder tree to be d, then, we get the class of plane d-ary tree. This is because of the fact that in this case the number of internal nodes is correlated to leaves.*

3.5.4 Increasing trees

A labelled tree is increasing when all the paths from the root to a leaf are (strictly) incresing. [3]

Different families of increasing trees can be considered. Even if the name *'increasing trees'* is classically used to denote a more specific families of trees.

Definition 3.5.18. *Let $\phi(u)$ be a colored degree function as defined in Definition 3.5.6. The class of* ***increasing trees*** *parametrized by $\phi(u)$ contains all trees that are labelled and rooted. Let n be the number of nodes in the tree. Then the label on the nodes in the tree goes from* 1

[3] In fact Strict increasing trees would have been a more accurate name but the name increasing trees has become widely known. In this thesis the distinction is important since we will consider some families of trees that are not necessarily strictly increasing.

to n. Additionally, all paths from the root to a leaf are strictly increasing. The size of a tree is the total number of nodes.

A consequence from the definition is that *increasing trees* have no label repetitions. Flajolet, Bergeron and Salvy in [**BFS92**] introduced a systematic study of increasing trees in terms of asymptotics and parameters on these trees. It should be noted that other families of increasing trees have also beend described in the literature such that the ones studied in [**JKP11, KP16**].

It is possible to construct plane or non-plane varieties of trees depending on how we interpret the *colored degree function*

$$\mathcal{T} = \mathcal{Z}^{\square} \star \underset{\Omega}{\mathsf{Seq}}\,(\epsilon + T) \qquad T = \mathcal{Z}^{\square} \star \underset{\Omega}{\mathsf{Set}}\,(\epsilon + T)$$

Where Ω is a multiset of of integers defined as in Section 3.5.1. If we denote

$$\phi(u) = \sum_{\omega \in \Omega} u^{\omega}.$$

The operators Seq_{Ω} and Set_{Ω} can be defiend as follows:

$$\underset{\Omega}{\mathsf{Seq}}(\mathcal{A}) = \sum_{k \geq 0} \phi_k \,\underset{k}{\mathsf{Seq}}(\mathcal{A}).$$

The set operators on Ω can be defined similarly.

So that for the Seq operator:

$$\phi(u) = \sum_{\omega \in \Omega} u^{\omega},$$

and for the Set operator:

$$\phi(u) = \sum_{\omega \in \Omega} \frac{u^{\omega}}{u!}.$$

The specifications translate to the following integral form

$$T(z) = \int_0^z 1 + \phi(T(u))\, du. \tag{3.5}$$

An example of the plane scheme is given by

Example 3.5.19. *Increasing binary trees. Take $\Omega = \{1, 1, 2\}$ which gives $\phi(u) = 2u + u^2$. The number of increasing binary trees of size n corresponds to the number of permutations referenced under EIS A000142. The trees of size 3 are depicted in Figure 3.13.*

Another example, this time corresponding to the non-plane scheme

Example 3.5.20. *Recursive trees. Take $\Omega = \{1, 2, \dots\}$ which gives $\phi(u) = \mathrm{e}^u - 1$. The number of recursive trees of size n corresponds also to the number of permutations but of size $n - 1$.*

Increasing trees can often be described as the result of a dynamic construction: the nodes are added one by one at successive integer-times in the tree (their labels being the time at which they are added). This description sometimes allow to apply probabilistic method to prove theorems about some characteristics such as the height of the tree, and it also often gives a

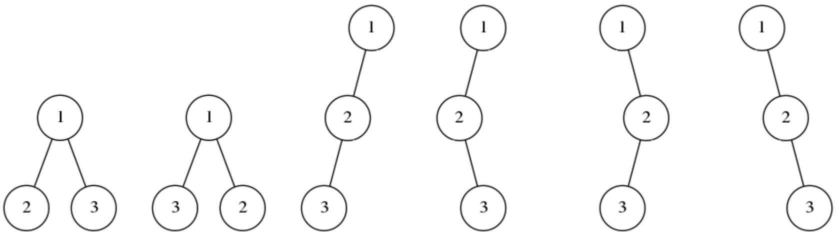


Figure 3.13: All 6 increasing binary trees of size 3.

very efficient way to generate large trees from the considered class using simple, iterative and local rules.

3.5.5 Incremental process for increasing trees

In this section we present an *incremental processes* for two models of increasing trees and for *binary search trees*. The idea we want to build a random tree of the desired class such that when we stop the process and have a tree of size n. All trees of size n are equiprobable in other words the underlying distribution is the *uniform distribution*. It is not possible for all varieties of increasing trees to admit a construction using successive insertions see [**PP07**] for more details.

We illustrate this incremental process on three classes of trees: *recursive trees*, *increasing binary trees* and *binary search trees*.

Recursive trees are a simple model of trees which were originally designed as a simple model for the spread of epidemics [**Moo74**]. Combinatorially, a recursive tree is a rooted non-plane (i.e. the order of siblings is irrelevant) tree whose nodes are labelled from 1 to the number of nodes in such a way that each label appears exactly once, and the labels increase along all branches. We denote by $\mathcal{R}_n$ the class of all n-node recursive trees. Now, consider a sequence of random trees $(t_n)_{n\geq 1}$ built recursively as follows: t_1 has only one node, labelled by 1. Given t_{n-1}, attach a new child labelled by n to a node picked uniformly at random among the $n-1$ nodes of t_{n-1}. Then, it is known that for all $n \geq 1$, t_n is a tree taken uniformly at random in $\mathcal{R}_n$, the set of all n-node recursive trees.

Both analytic combinatorics and probabilistic methods, as well as a bijection with permutations, have been used to understand the typical shape of a large recursive tree: it is known that the degree of the root grows as $\ln n$ (see [**Drm09**, Sec. 6.1]), the height as $c \ln n$ (for an explicit constant c – see [**Pit94**]), the proportion of nodes of arity $k \geq 0$ converges to 2^{-k} (see [**Drm09**, Th. 6.8]).

Incresing binary trees are rooted plane binary trees. Internal nodes are labelled from 1 to n such that along each branch of the tree, the labels are strictly increasing. We denote by $\mathcal{B}_n$ the class of all node binary increasing trees. The evolution process that builds this class of trees is defined by the follwing. Consider a sequence of random trees $(g_n)_{n\geq 1}$ built recursively as follows: g_1 has a single internal node labelled 1 and two leaves attached to it. Given g_{n-1}, replace a leaf picked up uniformly at random with a new node labelled n that has two leaves attached to it. Then, g_n is a tree which contains n internal nodes. It is known that for all $n \geq 1$, g_n is a tree taken uniformly at random in $\mathcal{B}_n$.

Binary search trees form an important and classical data structure. The structure is efficient especially for inserting and searching elements. It is also simple enough to implement. There are many existing studies on this class of trees studied from different point of views, see for instance [**FGM97**, **Mah92**]. We describe the evolution process of these trees in the following. We generate a random tree t_n of size n as follows. Suppose we are given a random permutation σ of n elements. At first, t is reduced to the trivial tree consisting of a single root labelled σ_1. Then, given t_{n-1}, we compare σ_n with the root labelled σ_1. If $\sigma_n < \sigma_1$, then descend into the left subtree, otherwise into the right subtree. Continue with the root of the chosen subtree as current, according to the same rule. Finally, attach a leaf labelled σ_n at the first empty place.

It has been shown in various places [**Drm09**, **MTS18**, **SM01**] that the two models of *binary search trees* and *increasing binary trees* are equivalent in the sense that the underlying tree-shapes (the one obtained after removing all labels) have the same probability distribution.

3.5.6 Monotonic trees

Prodinger and Urbanek in a paper entitled 'On monotone functions of tree structures' [**PU83**] introduced a property of a tree labelling.

Definition 3.5.21. *From* [**PU83**]*, Let T be a rooted labelled tree with n nodes such that the labels belong to $\{1, \dots, k\}$ with $k \leq n$. The function f gives the label of each node. The labelling function is called* ***monotone*** *if whenever a node x is a son of a node y, then $f(y) \geq f(x)$.*

If $k = n$ they call it a ***monotone bijection***. As we see from the definition there is no requirement for the labelling to start at 1.

There exists some research on *monotonic trees* that study some paramters on these trees such that [**MP05**, **Kir84**, **Kem93**, **Bli87**] and the thesis of of Morris [**Mor04**]. Some more recent studies include models of *multilabelled increasing trees* by Panholzer and Kuba in [**KP16**].

In Chapter 4, Chapter 5 and Chapter 7 we will be working on trees that are labelled and such that the underlying structure is the one of *Schröder trees*. Then, we make some definitions.

Definition 3.5.22. *A labelled Schröder tree, has the skeleton of a Shröder tree (see Section 3.5.3) and is such that only internal nodes have integer labels.*

From the definition we see that the leaves of a labelled Schröder tree are not labelled. Therefore,

Definition 3.5.23. *A* ***monotonic Schröder tree*** *is a labelled Schröder tree, such such that the root node has label 1 and along each branch the labellings are weakly increasing and if m is the largest integer of the tree then all labels from 1 to m appear.*

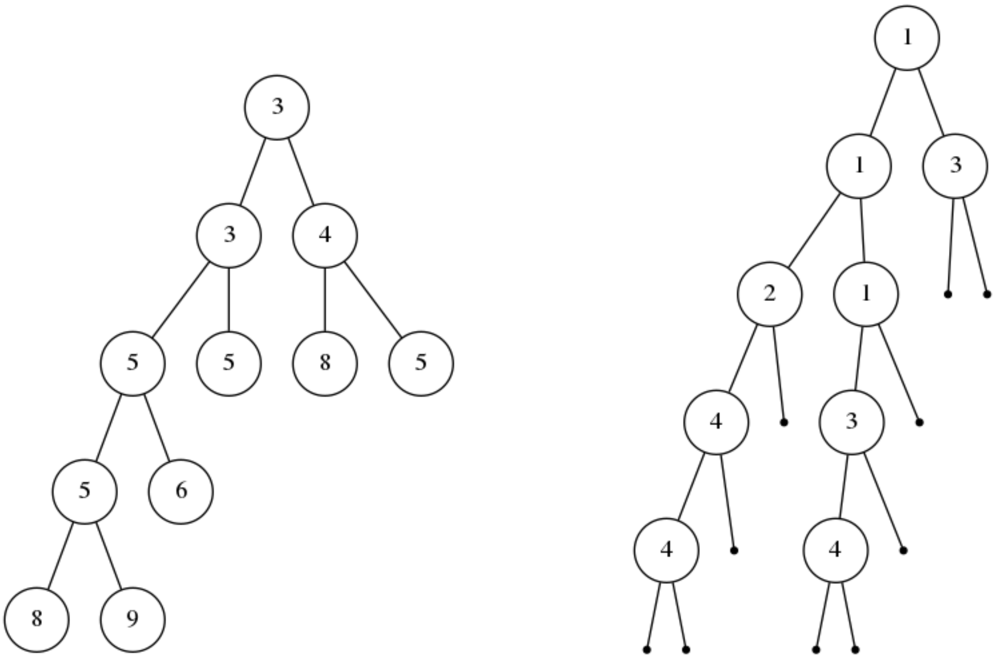


Figure 3.14: (left) A monotonic tree in the sense of [**PU83**] with $k = 9$, (right) A monotonic tree in the sense of Definition 3.5.23 the root always starts with label 1 and there are no skipped labels.

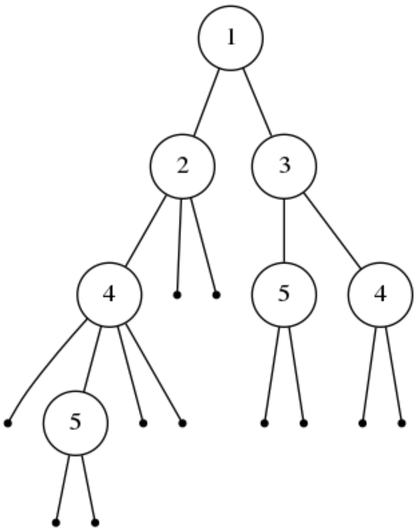


Figure 3.15: An example of a strict monotonic Schröder tree seen in Definition 3.5.24 (the labels are strictly increasing along all paths) it is also a monotonic tree.

Definition 3.5.24. *A **strict monotonic Schröder tree**, is a monotonic Schröder tree, such that along each branch the labellings are strictly increasing.*

monotonic Schröder trees and *strict monotonic Schröder trees* allow for a label appearing multiple times. However in *strict monotonic Schröder trees* the labels appear in different branches of the tree, whereas in *monotonic Schröder trees*, they can appear in the same branch. See Figure 3.15 for an example of a *strict monotonic Schröder tree*, however, the right tree on Figure 3.14 is not strict monotonic.

Remark 3.5.25. *Our use of monotonic labelling in Definition 3.5.23 and Definition 3.5.24 is connected to the fact that different nodes with the same label can appear in the tree as long as they do not violate the condition for monotonicity. However, in all our models the root always starts with label* 1*. Therefore, we do not need to fix a parameter* k *as in* [**PU83**].

We end this section with a final definition

Definition 3.5.26. *A* ***connected monotonic Schröder tree*** *is a labelled Schröder tree such that the labellings are weakly increasing along branches and when a label* i *appears for the first time in the tree on node* v *(*v *is also the closest node to the root labelled* i*), then all other occurences of* i *appear in subtree of* v*.*

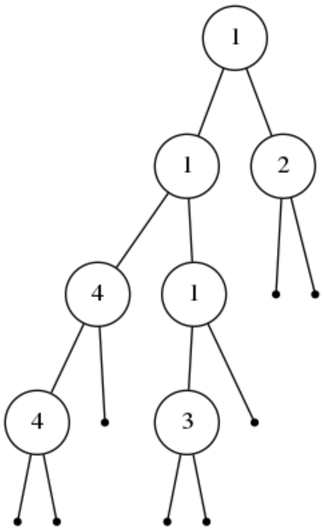


Figure 3.16: An example of a connected monotonic Schröder tree seen in Definition 3.5.26 (the same labels are connected).

As a result of this definition we see that when a label i appears in the tree, all other occurences of i are connected. For instance, the tree in Figure 3.15 is not connected because the label 4 appears on different branches. Whereas the tree in Figure 3.16. As a result we see that a connected monotonic Schröder tree is a monotonic Schröder tree but not a strict monotonic Schröder tree in general (since in the latest the same label can appear on different branches).

CHAPTER 4

Three models of increasing Schröder trees

Contents

> what appears to be different truths are like apparently different countless leaves of the same tree.
>
> *Mohandas Karamchand Gandhi (1869 - 1948), Teachings of Mahatma Gandhi*

4.1 Introduction

The aim of this chapter is to introduce new combinatorial models for phylogenetic trees: the main idea is to add node labels in order to encode chronology in the classical model of trees introduced by Ernst Schröder in 1870 in the seminal paper *Vier Combinatorische Probleme* [**Sch70**] and presented in Section 3.5.3.

In his paper (see the second problem), Ernst Schröder introduces a simple model of *phylogenetic tree model*, and enumerate this class of trees by their number of leaves.

In biology, a phylogenetic tree is a classical tool to represent the evolutionary relationship among species. At each branching node of the tree, the descendant species from distinct branches have differentiated themselves in some manner and are no more dependent: the past is shared but the futures are independent.

The main limitation of Schröder's model of phylogenetic trees is that it does not take into account the chronology between the different branching nodes. Since then, probabilistic approaches have been developed to consider this chronology: in particular in the context of binary trees, one can mention, e.g., the stochastic model of Yule [**EY25**] and its generalisation by Aldous [**Ald96**].

However, to the best of our knowledge, there seems to have been no attempt to combinatorially enrich Schröder's original model in order to encode the chronology of evolution.

To do so, we consider labelled versions of *Schröder trees*, where the labels represent the order at which branchings occur. In Figure Figure 4.1 we have represented on the left hand-side a classical Schröder trees of size 50 (i.e. with 50 leaves), and, on the right hand-side, a labelled version of the same tree: time is on the vertical axis, from top to bottom, and a node of label x is placed at time x (the horizontal placement is irrelevant).

Discussion of related models: Increasing trees are classical in the literature of combinatorics and biology. Since they simulate *evolution processes*: for example, Bergeron, Flajolet and Salvy [**BFS92**] studied several families of increasingly-labelled trees (see Section 3.5.4 for more details), and, to do so, they developed some tools that are now classical in *analytic combinatorics*. As an example, one of these classical tools is the integration of the Greene operators. We refer the reader to [**Drm09**] where more recent results on various families of increasing trees and the analytic combinatorics methods to quantitatively study them are surveyed.

However, the nature of our problem combines the models of *Schröder trees* and that of *Increasing trees*. From one side, the size notion of the trees is their number of leaves which is total number of living individuals. From the other side, we want that internal nodes be labelled increasingly to denote the appearance time differentiation.

Although our three models of increasing Schröder trees are more involved, our proofs rely on the same three methods used in the literature for the recursive trees: analytic combinatorics, a dynamical evolution and probabilistic methods, and bijections with classes of permutations.

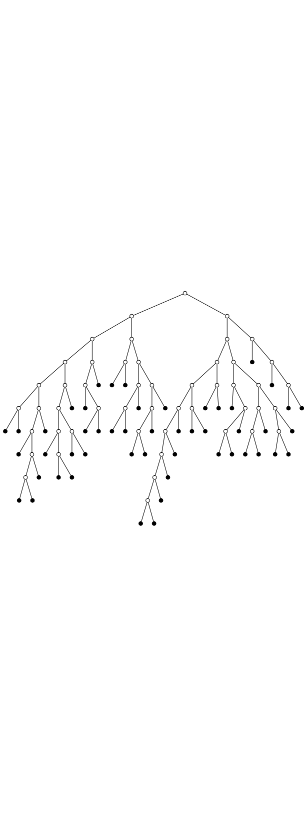


Figure 4.1: A Schröder tree: without chronological evolution (on the left-hand side), and with chronological evolution (on the right-hand side): the label of a node is represented as the distance from this node to the root.

Our main contributions: Although, as mentioned above, many variations of the recursive tree have been studied, this chapter together with the pair of papers: the long [**BGMN20**] and short version [**BGN19**] contains the first studies of increasing versions of the classical model of Schröder. We aim at defining an evolution process associating to a given Schröder tree structure an evolution represented by an increasing labelling of its internal nodes. Furthermore we also focus on relaxing the labelling constraints by allowing repetitions of labels. In the dynamical construction of the trees, allowing repetition of labels mean allowing adding several nodes at once in the tree. Our generalisations can be seen as natural discrete-time versions of the classical probabilistic model of Yule trees (see, e.g., [**SM01**]) where the time between two branchings are exponentially distributed.

This work is a part of a long-term overarching project, in which we aim at relaxing the classical rules of increasing labelling (described in, e.g., [**BFS92**]), by, for example, allowing labels to appear more than once in the tree. The following papers are part of this strand of research: [**BGGW20, BGNS20**] introduce and study models of label trees with less-constrained increasing labelling rules, but also other graphs structures like [**BDF$^+$16**] focuses on increasingly-labelled “diamonds” and [**GGKW20**] on a compacted structure that specifies classes of directed acyclic graphs.

In this chapter, we introduce three new different models of Schröder trees with chronological evolution: the *increasing Schröder trees*, the *strict monotonic Schröder trees* and the *strict monotonic general trees*. One important feature of these models is that they can all be simulated efficiently as will be seen in Chapter 7. The first two models are based on some *increasingly labelling* of Schröder trees, repetition of labels is allowed in the second model. In the last model increases we increase the expressivity by allowing a new type of internal nodes. For all of the three models, we prove asymptotic results about important characteristics of a typical large tree of this class (e.g. root distribution, number of nodes of arity 2, 3, etc, height of the tree, etc – see Table 4.1 where our main results are summarised), and design an algorithm that generates a large tree taken uniformly at random among all trees of a given size in the class. The quantitative analysis of the three models and the design of the random samplers rely on a combination of analytic combinatorics methods (see [**FS09**] for a survey), probabilistic methods (in particular methods developed by Devroye [**Dev90**] to study the height of *split trees*), and bijective methods (we exhibit bijections between our classes of trees and classes of permutations, these are then useful for the analysis of different characteristics and for the design of the generation algorithms). In particular, we exhibit interesting relations between Stirling numbers and parameters on trees such that the labelling of nodes, the number of internal nodes, and the depth of a leaf.

Generic approach highlighted in the chapter: Similarly to the recursive tree, all of our three models have a *generic constrained evolution process*. The specificity of each model is induced by small changes of the evolution process: we give here a generic, non precise description of the evolution process, details specific to each family of trees will be detailed in each section:

- *Start with a single (unlabelled) leaf;*
- Iterate the following process: *at step ℓ (for $\ell \geq 1$), select a subset of leaves and replace each selected leaf by an internal node with label ℓ attached to an arbitrary sequence of leaves.*

Note that the increasing labelling corresponds to the chronology of the construction of the tree: internal nodes labelled by an integer ℓ were added at time ℓ. Our three models differ from each other by different constraints on the set of selected leaves: in our first model, this subset is always of size 1, while it can be bigger in the other two models. The difference between our second model and third model is that internal nodes have arity at least 2 on the second model, while they can have arity 1 in the third model. Importantly, in all three models our Schröder trees can be seen as phylogenetic trees of n species (n being the number of leaves): the labels of internal nodes stand for the times at which different branches of the phylogenetic trees split.

	Number of trees	Distinct labels	Internal nodes	Depth LM leaf	Height
Increasing Schröder trees	$n!/2$	$n - \ln n$	$n - \ln n$	$\ln n$	$\Theta(\ln n)$
Strict monotonic Schröder trees	$(n-1)!/(2\,(\ln 2)^n)$	$0.72\,n$	$n - \ln 2\ \ln n$	$\ln n$	
Strict monotonic general trees	$c\,(n-1)!\,2^{(n-1)(n-2)/2}$	$\Theta(n)$	$\Theta(n^2)$	$\Theta(n)$	$\Theta(n)$

Table 4.1: Summary of the main analytic results of this chapter: behaviour of the characteristics of a large typical tree of each of the three classes of labelled Schröder trees. The parameter n stands for the size of the trees (i.e. their number of leaves) and the results are asymptotic when $n \to +\infty$. (LM stands for "leftmost" and c is a positive constant.)

Plan of the chapter: Each of the three main parts (Section 4.2,Section 4.3 and Section 4.4) is dedicated to one of our three new models of labelled Schröder trees. The organisation inside each section is similar: after defining the model we show theorems about different characteristics of the trees using analytic combinatorics and bijective methods. We then exhibit the associated dynamical evolution that generates the considered class of trees, and use this evolution process to (a) design an efficient random sampler for this class of trees and (b), in some cases, to prove some probabilistic results about the height of a typical large tree from this class.

4.2 Increasing Schröder trees

The first model we are interested in is a generalisation of the Schröder tree, a classical combinatorial structure that was originally introduced in the context of phylogenetics [**Sch70**].

Our generalisation consists in labelling the internal nodes of a Schröder tree – denote by ℓ their number – with the integers $\{1, \dots, \ell\}$ with the constraints that each label appears exactly once and a node's label is larger than the label of its parent; such a labelling of a tree is called "increasing", we call such a constrained-labelled Schröder tree an *increasing Schröder tree*. In the tree seen as an evolutionary process, the labels can be interpreted as the order of appearance of the different nodes (which, for example, stand for different species). Several classes of increasingly-labelled trees have already been studied in the literature using *analytic combinatorics* [**FS09**] methods, but these methods applied to the Schröder tree would raise important technical problems. The novelty of our approach is to use a dynamical description of the increasing Schröder tree inspired by its evolutionary interpretation; this allows us to give the first analytical results about this combinatorial structure.

4.2.1 The model and its context

We define rooted trees as genealogical structures: the root is the unique common ancestor of all nodes of the tree, each node except the root has exactly one parent (the root has no parent), nodes that have no children are called *leaves*, nodes that have at least one child are called *internal nodes*. The *arity* of a node is it's number of children. We say that a tree is *plane* if siblings (nodes that have the same parent) are ordered.

We are interested in an increasingly-labelled variation of Schröder trees that were presented in Section 3.5.3.

Definition 4.2.1. *An* increasing Schröder tree *has a Schröder tree structure and its internal nodes are labelled with the integers between* 1 *and* ℓ *(where* ℓ *is the number of internal nodes) in such a way that each label appears exactly once and each sequence of labels in the paths from the root to any leaf is (strictly) increasing.*

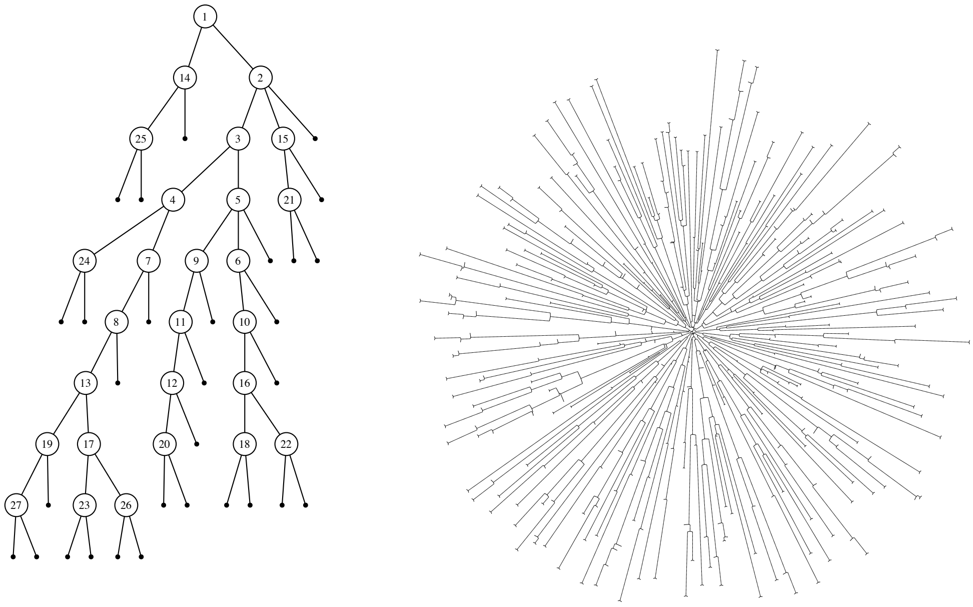


Figure 4.2: Two increasing Schröder trees

Increasing trees can, to a certain extent, be specified using the Greene operator $^{\square}\star$ (see, for example, [FS09, page 139]), and the specification can then be translated into an equation satisfied by the exponential generating function of the increasing tree class. Since in our context the size of a tree is the number of its leaves while only internal nodes are labelled, we need to introduce a second variable u to mark the internal nodes. Let us denote by $s_{n,\ell}$ the number of increasing Schröder trees with n leaves and ℓ internal nodes. Following standard methods in analytic combinatorics we define a generating function that is ordinary for the leaf marks and exponential for the internal node marks: we set $S^*(z,u) = \sum_{n,\ell} s_{n,\ell}\, z^n\, u^\ell/\ell!$. The specification of this combinatorial class is

$$\mathcal{S}^* = \mathcal{Z} \cup \mathcal{U}^{\square} \star \mathrm{Seq}_{\geq 2}\, \mathcal{S}^*.$$

Using the symbolic method, we obtain the following equation satisfied by $S^*(z,u)$:

$$S^*(z,u) = z + \int_{v=0}^{u} \frac{S^*(z,v)^2}{1 - S^*(z,v)} \mathrm{d}v.$$

Although this integral equation could be analysed further in order to get information about increasing Schröder trees, this analysis would be very cumbersome; a better approach is to see the Schröder tree as the result of an evolutionary process. Another advantage of this new approach is that it extends to other families of labelled Schröder trees for which there seems to be no (classical) specification, even using the Greene operator: one such example is the family of *strict monotonic Schröder trees* studied in Section 4.3.

In Figure 4.2 we have represented two increasing Schröder trees: both are generated uniformly at random among all increasing Schröder trees of the same size: size 30 on the left, size 500 on the right. The left-hand-side tree has 27 internal nodes (and 30 leaves). It is the same tree as the one represented in Figure 4.1, where its chronological evolution is represented on the right-hand side: the internal node labelled by ℓ is displayed on level $\ell-1$ (i.e. at distance $\ell-1$ from the root on the vertical axis), for all $\ell \in \{1, \dots, 27\}$. The right-hand-side one is drawn using a circular representation, which is often used for phylogenetic trees: the labels are omitted but as in Figure 4.1, the length of an edge is proportional to the difference of the labels of the two nodes it connects. This right-hand-side tree has 492 internal nodes (and 500 leaves).

We now introduce an evolution process generating *increasing Schröder trees*:

- *Start with a single (unlabelled) leaf;*
- Iterate the following process: *at step ℓ (for $\ell \geq 1$), select one leaf and replace it by an internal node with label ℓ attached to an arbitrary sequence of new leaves.*

Recall that we define the size of a Schröder tree to be its number of leaves. It is important to note that, because a Schröder tree with $n-1$ internal nodes has at least n leaves, the evolution process defines a bijection between the set of all n-leaf Schröder trees and the set of all sequences $(d_\ell^{(n)}, u_\ell^{(n)})_{1\leq \ell<n}$ such that for all $1 \leq \ell < n$, $u_1^{(n)} = 1$, $d_\ell^{(n)} \geq 2$, $1 \leq u_{\ell+1}^{(n)} \leq \sum_{i=1}^{\ell} d_i^{(n)} - (\ell-1)$, and $\sum_{i=1}^{\ell} d_i^{(n)} = n$.

By taking all trees of the same size together, we obtain the following induction equation, enumerating increasing Schröder trees by size: if, for all $n \geq 0$, t_n is the number of n-leaf Schröder trees, then $t_1 = 1$ and, for all $n \geq 2$,

$$t_n = \sum_{\ell=1}^{n-1} \ell \, t_\ell. \tag{4.1}$$

4.2.2 Overview of the main results

After solving the counting problem of ***increasing Schröder trees***, we show how trees of size n can be constructed from trees of size $n-1$ in Section 4.2.4. Then, in Section 4.2.5 we exhibit different bijections with sub-classes of *permutations* in Section 4.2.3. In fact these bijections give important information about the tree structure. For instance, we will show that the number of internal nodes of an Increasing Schröder tree is related to the number of *cycles* in permutations by a very simple formula, see Theorem 4.2.8. The last bijection will show another interesting result. In fact, the number of cycles in a permutation is also linked to the depth of the leftmost leaf in the tree as shown in Theorem 4.2.12. In Section 4.2.6, we show the relationship between the trees and runs in permutations. Finally, Section 4.2.7 is dedicated to the study of some parameters on these tree that are summarised in Table 4.2.

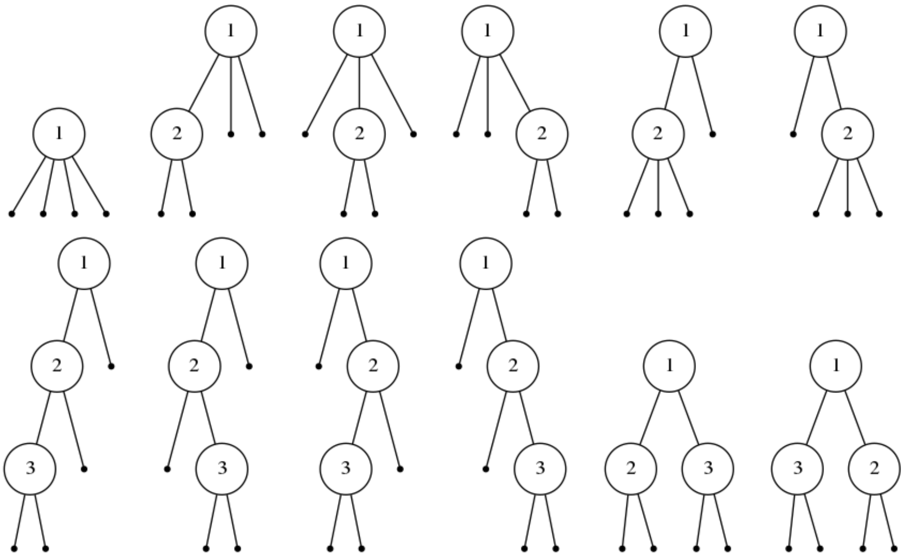


Figure 4.3: All *increasing Schröder trees* of size 4

	Mean	Variance	Limit law
Internal nodes	$n - \ln n$	$\ln n$	Normal
Number of binary nodes	$n - 2\ \ln n$	$4\ \ln n$	Normal
Number of ternary nodes	$\ln n$		
Depth of the leftmost leaf	$\ln n$	$\ln n$	Normal
Height of the tree	$\Theta\left(\ln n\right)$		
Degree of the root	$2\,\mathrm{e} - 3$	$14\,\mathrm{e} - 4\,\mathrm{e}^2 - 8$	modified Poisson
Leaves attached to the root	$\frac{2\,\mathrm{e}}{n}$	$\frac{2\,\mathrm{e}}{n}$	

Table 4.2: Summary of the main results on parameters of *Increasing Schröder* trees

4.2.3 Exact enumeration and relationship with permutations

Let $\mathcal{T}$ denote the class of increasing Schröder trees. Using the evolution process, we get the following specification for $\mathcal{T}$:

$$\mathcal{T} = \mathcal{Z} \cup \left(\Theta\mathcal{T} \times \mathrm{Seq}_{\geq 1}\,\mathcal{Z}\right). \tag{4.2}$$

In this specification, $\mathcal{Z}$ stands for the leaves, and the operator Θ is the classical pointing operator (see [**FS09**, page 86] for details). The specification is a direct rewriting of the evolution process: a tree is either of size 1 ($\mathcal{Z}$), or it has been built by pointing a leaf in a smaller tree ($\Theta\mathcal{T}$) and replacing it by a sequence of at least two leaves. Although the latter sequence is of length at least 2, we use the operator $\mathrm{Seq}_{\geq 1}(\mathcal{Z})$ instead of $\mathrm{Seq}_{\geq 2}(\mathcal{Z})$ because the leaf that was pointed is reused as the leftmost child of the new internal node.

The symbolic method translates this specification into a functional equation satisfied by the generating function associated to the combinatorial class of increasing Schröder trees. Note that although the increasing Schröder trees are labelled, this labelling is transparent, i.e. it is possible to work with ordinary generating functions (as opposed to exponential generating functions). This is because the size of an increasing Schröder tree is its number of leaves, and the leaves are not labelled. We define the *ordinary generating function* associated to $\mathcal{T}$ by $T(z) = \sum_{n\geq 1} t_n\, z^n$, where t_n is the number of increasing Schröder trees of size n. Using the symbolic method (in particular, pointing at a leaf translates into a differential operator), we get

$$T(z) = z + \frac{z^2}{1-z}\, T'(z). \tag{4.3}$$

Writing $(1-z)T(z) = z(1-z) + z^2 T'(z)$ and extracting the n-th coefficient on both sides of this equation, we get that,

$$t_n = \begin{cases} 1, & n = 1, 2 \\ n\, t_{n-1}, & n \geq 3 \end{cases} \tag{4.4}$$

Then, the first values of t_n are

$$(t_n)_{n\geq 0} = 0, 1, 1, 3, 12, 60, 360, 2520, 20160, 181440, 1814400, \ldots$$

Theorem 4.2.2. *The number of trees of size n for all $n \geq 2$,*

$$t_n = \frac{n!}{2}.$$

for all $n \geq 3$, $t_n = n\ t_{n-1}$: we get back the recurrence exhibited earlier in Equation (4.1). Using the fact that $t_1 = t_2 = 1$, we get that $t_n = {}^{n!}/_{2}$ for all $n \geq 2$. Note that the radius of convergence of the ordinary generating series $T(z)$ is 0; this series is thus purely formal.

In Figure 4.3 we draw all trees of size 4. The sequence of numbers t_n appears in EIS A001710 as the order of the alternating group Alt_n, or the number of even permutations of n letters (see Section 3.1 for more details on this group).

4.2.4 Iterative construction of a tree

From Equation (4.4) we see that a tree of size n can be constructed from the set of trees of size $n-1$.

Proposition 4.2.3. *Each increasing Schröder tree of size n, $t \in \mathcal{T}_n$ can be constructed from a tree $t' \in \mathcal{T}_{n-1}$ uniquely as follows,*

- *Either by choosing one of $n-1$ leaves of t' and replace it with a binary node labelled with the successor of the highest integer already appearing in the tree.*

- *Or by adding a new rightmost leaf to the last internal node (i.e the internal node with highest label) that appears in t'.*

Proof. By induction. For $n = 2$ there is only one tree. For $n \geq 3$, from Equation (4.4) there are exactly n way to add a leaf to a tree of size $n - 1$ so that it becomes a tree of size n. Therefore if the above construction is unambiguous and all the constructed trees belong to $\mathcal{T}_n$ the result holds. We see that we put a binary node to replace one of the $n-1$ leaves. Since the trees are plane this construction is not ambiguous. Finally when adding a new rightmost leaf to the last internal node added to the tree is also unambiguous. In summary, at each iteration step we either create a binary node or add a new leaf to a specific node, and since we have no limit on the degree of a node the construction gives a tree of $\mathcal{T}_n$. □

In the following we will exhibit three bijections with sub-classes of permutations based on this iterative construction and we will obtain nice bijective results. In particular the reader can see Corollary 4.2.9, Corollary 4.2.13, Corollary 4.2.14 and Theorem 4.2.20.

4.2.5 Bijections with permutations and relationship to internal nodes and depth of a leaf

4.2.5.1 Bijection with restricted permutations

As we saw in Section 4.2.4, at each iteration step if we have a tree of size n (respectively an n-permutation or a permutation of n elements), we have $n + 1$ ways of building a tree of size $n + 1$ (respectively an $(n + 1)$-permutation). We have n different ways of adding a binary node and one way of adding a new leaf to the last internal node. This corresponds to the $n+1$ places to add a new element in an n-permutation. In this bijection we fix the position of the element that adds a new arity to the tree in the permutation.

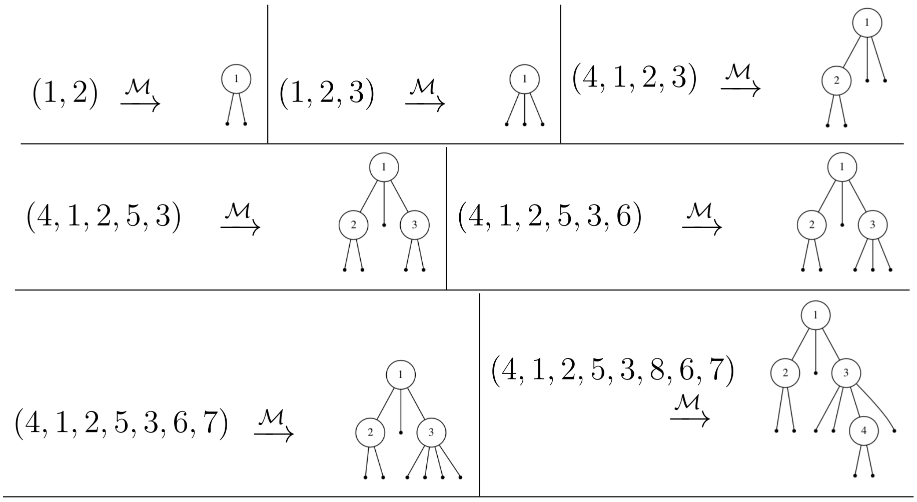
The fact that the number of increasing Schröder trees of size n is equal to $t_n = \frac{n!}{2}$ hints at the existence of a relationship between our model of increasing trees and a subclass of permutations. In this section, we aim at exhibiting this relationship. In this section we will exhibit two bijections. The idea is to put a restriction on the number of permutations so that they become equal to $\frac{n!}{2}$. Maybe one of the simplest way to think of it is to put a restriction that integer one appears before two in the permutation (i.e its index is smaller). However a more elaborate bijection will be presented that makes use of another restriction namely the one of cycles and this last restriction on cycles will be of great interest to us since some parameters on trees can be seen directly on permutations.

We will denote by $\sigma = (\sigma_1, \ldots, \sigma_n)$ the size-n permutation that sends i to $\sigma_i \in \{1, \ldots, n\}$ for all $i \in \{1, \ldots, n\}$. For all $k \in \{1, \ldots, n\}$, we denote by $\sigma^{-1}(k)$ the preimage-image of k by σ, and sometimes call $\sigma^{-1}(k)$ the “position” of k in the permutation σ.

We define recursively a map $\mathcal{M}$ between $\mathcal{HP}$, the class of permutations such that 1 appears before 2, and the class $\mathcal{T}$ of increasing Schröder trees.

The only element of $\mathcal{HP}$ of size 2 is the permutation $(1, 2)$; we set its image to be the tree whose root is labelled by 1 and has two (unlabelled) leaf-children. Now assume that we have defined $\mathcal{M}(\sigma)$ for all permutations $\sigma \in \mathcal{HP}$ of size at most $n-1$ for some $n \geq 2$ and let σ be a size-n permutation in $\mathcal{HP}$. We distinguish two cases according to the preimage of n by σ; we denote by $\hat{\sigma} = \sigma \backslash n$, the permutation obtained by removing the largest element in σ. For example, if $\sigma = (4,1,5,2,3)$, then $\hat{\sigma} = (4,1,2,3)$; $\hat{\sigma}$ can be seen as the permutation induced by σ on $\{1,\ldots,n-1\}$.

- If $\sigma_n = n$ then, we set $\mathcal{M}(\sigma)$ to be the tree $\mathcal{M}(\hat{\sigma})$ in which we add a new rightmost leaf to the internal node with the largest label.
- If $\sigma_n = k < n$, then, we build $\mathcal{M}(\sigma)$ as follows: create a new binary node ν labelled with the smallest integer that does not appear as a label in $\mathcal{M}(\hat{\sigma})$ and attach two new leaves to this internal node. Insert this tree in $\mathcal{M}(\hat{\sigma})$ by placing ν in the k-th leaf (we assume, for example, that the leaves are ordered in the depth-first order) of $\mathcal{M}(\hat{\sigma})$.



$(1,2) \xrightarrow{\mathcal{M}}$
1
$(1,2,3) \xrightarrow{\mathcal{M}}$
1
$(4,1,2,3) \xrightarrow{\mathcal{M}}$
1
2
$(4,1,2,5,3) \xrightarrow{\mathcal{M}}$
1
2
3
$(4,1,2,5,3,6) \xrightarrow{\mathcal{M}}$
1
2
3
$(4,1,2,5,3,6,7) \xrightarrow{\mathcal{M}}$
1
2
3
$(4,1,2,5,3,8,6,7) \xrightarrow{\mathcal{M}}$
1
2
3
4

Figure 4.4: A size-8 example of the mapping $\mathcal{M}$

In Figure 4.4 we present the mapping on an example. Remark that we have ordered the steps reversely to understand the process in a constructive way.

Theorem 4.2.4. *The map $\mathcal{M}$ is a one-to-one correspondence between $\mathcal{HP}$ and $\mathcal{T}$.*

Proof. First note that the image by $\mathcal{M}$ of a permutation of size n is a Schröder tree of size n: indeed, at each iteration we remove exactly one element from the permutation and add exactly one leaf to the tree by either adding a leaf to the node with largest label or by removing one leaf and adding two new ones. Since the number of permutations of size n in $\mathcal{HP}$ is equal to the number of Schröder trees of size n, it is enough to prove that $\mathcal{M}$ is injective to conclude the proof. The mapping is injective since it corresponds to the construction in Proposition 4.2.3. □

4.2.5.2 Bijection with cycles in permutations and relationship to internal nodes

The idea here is to present a bijection with permutations based on their number of cycles. As we said in the start of this Section 4.2.5.1. We need a restriction on the set of permutations.

Let us define the combinatorial class $\mathcal{PR}$ (permutations with cycle restriction) to be the set of permutations in which the elements 1 and 2 belong to different cycles.

Lemma 4.2.5. *For $n \geq 2, PR_n = \frac{n!}{2}$.*

Proof. This can be seen by recurrence. For $n = 2$ there is one such permutation namely $(1)(2)$. Then if PR_n is the set of n-permutation with elements 1 and 2 in different cycles. $PR_{n+1} = (n+1)PR_n$. Since from the set $\mathcal{PR}_n$ a new element can be added to any place in the different cycles (the number of places is n) or create a new cycle. Solving the recurrence we get $\frac{n!}{2}$. □

In this section, let σ be an n-permutation with k cycles. Then we denote

$$\sigma = c_1 \circ c_2 \circ \cdots \circ c_k$$

as a product of cycles c_i. This notation has been defined in Section 3.1.1. A *cycle* can be ordered canonically by putting the smallest integer in the beginning. Let $c = (a_1, a_2, \ldots, a_i)$ be a cycle that contains i elements. A new element e can be added in i different places. We add the new element to the right of the element at the selected index, that is, let $1 \leq j \leq i$. Then adding e in place j we obtain a new cycle $c' = (a_1, \ldots, a_j, e, a_{j+1}, \ldots, a_i)$. We also denote by $c_{i,j}$ the element of position j in the cycle i.

Let us define also $\sigma^{-1}(k)$ the function that returns a pair of integers (i, j) where i is the cycle number that contains integer k and j the position of integer k in the cycle c_i. Finally let $|c_j|$ be the number of elements in the cycle c_j.

Finally we define $\sigma \backslash n$ to be the permutation σ from which the element n has been removed.

We will define the mapping $\mathcal{N} : \mathcal{T} \to \mathcal{PR}$ and show that

$$\mathcal{T} \overset{\mathcal{N}}{\cong} \mathcal{PR}$$

Definition 4.2.6. *We define the mapping $\mathcal{N}$ recursively as follow:*

- *If $\sigma = (1)(2)$ then $\mathcal{N}(\sigma)$ is the tree which is a binary root labelled* 1.
- *Else, Let $(i, j) = \sigma^{-1}(n)$ where n is the largest element in σ.*
  - *If $|c_i| = 1$ then, we set $\mathcal{N}(\sigma)$ to be the tree $\mathcal{N}(\sigma \backslash n)$ in which we add a new rightmost leaf to the last internal node of the tree (it is also the node with highest label).*
  - *Else, let $k = c_{i,j-1}$, we set $\mathcal{N}(\sigma)$ to be the tree $\mathcal{N}(\sigma \backslash n)$ in which a new binary node labelled ν with the smallest integer that does not appear as a label in $\mathcal{N}(\sigma \backslash n)$ and attach two new leaves to this internal node. Insert this binary node in $\mathcal{N}(\sigma \backslash n)$ by placing ν in the k-th leaf (we assume, for example, that the leaves are ordered in the depth-first order) of $\mathcal{N}(\sigma \backslash n)$.*

The mapping $\mathcal{N}$ might seem a bit complicated to explain in simple words but it is quiet simple and the Figure 4.5 shows the same tree as in Figure 4.4 but using this new bijection.

Theorem 4.2.7. *The map $\mathcal{N}$ is a one-to-one correspondence between $\mathcal{PR}$ and $\mathcal{T}$.*

Proof. The proof is very similar to the one in Theorem 4.2.4. At each step in the algorithm exactly one leaf is added to the tree (the total size is increased by 1). The corresponding behaviour on the tree for the last is unambiguous since it corresponds to a construction as in Proposition 4.2.3. Therefore the mapping $\mathcal{N}$ is also injective. □

This mapping is interesting since we can see directly deduce that the number of cycles is closely related to the number if internal nodes in the tree.

Theorem 4.2.8. *Let t be an increasing Schröder tree with n leaves and k internal nodes, and let $p = \mathcal{N}^{-1}(t)$. Let i be the number of cycles of p, then*

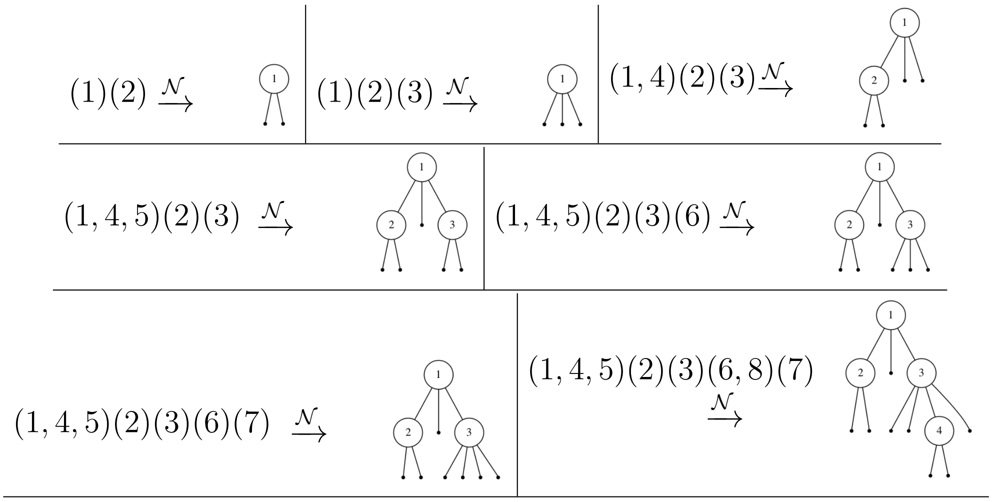
$$k = n + 1 - i.$$

Proof. In the mapping $\mathcal{N}$ we see that when adding a new element in the permutation. We have two options, either creating a new cycle, or adding the new element to an already existing cycle. In both cases the size of the final permutation is increased by one.

We also see that if we create a new cycle with the new element then we do not create any new internal nodes.

The trivial tree (a binary root labelled 1) has one internal node and its corresponding permutation has 2 elements and 2 cycles.

The maximum number of internal nodes for a tree of size n is $n - 1$ and its permutation has 2 cycles. If a tree has 3 cycles then its has $n - 2$ internal nodes and so on. The result follows straightly. □



$(1)(2) \xrightarrow{\mathcal{N}}$
1
$(1)(2)(3) \xrightarrow{\mathcal{N}}$
1
$(1,4)(2)(3) \xrightarrow{\mathcal{N}}$
1
2
$(1,4,5)(2)(3) \xrightarrow{\mathcal{N}}$
1
2
3
$(1,4,5)(2)(3)(6) \xrightarrow{\mathcal{N}}$
1
2
3
$(1,4,5)(2)(3)(6)(7) \xrightarrow{\mathcal{N}}$
1
2
3
$(1,4,5)(2)(3)(6,8)(7) \xrightarrow{\mathcal{N}}$
1
2
3
4

Figure 4.5: A size-8 example of the mapping $\mathcal{N}$

Corollary 4.2.9. *Let $t_{n,k}$ be the number of increasing Schröder trees of size n with k internal nodes. Let $PR_{n,i}$ be the number of n-permutations such that the elements 1 and 2 belong to different cycles and which have i cycles. Then,*

$$t_{n,k} = PR_{n,n+1-k}.$$

Proof. The result follows directly from Theorem 4.2.8. □

4.2.5.3 Another Bijection with cycles in permutations and relationship to the depth of the leftmost leaf

We present now a last bijection which is also based on the number of cycles in a permutation. The idea of this bijection looks like the previous one in Section 4.2.5.2 with small modifications. We define the mapping $\mathcal{O} : \mathcal{T} \to \mathcal{PR}$ and keep the same notations as in Section 4.2.5.2.

Definition 4.2.10. *We define the mapping $\mathcal{O}$ recursively as follow:*

- *If $\sigma = (1)(2)$ then $\mathcal{O}(\sigma)$ is the tree which is a binary root (a root with two leaves) labelled 1.*
- *Else, Let $(i, j) = \sigma^{-1}(n)$ where n is the largest element in σ.*
  - *If $|c_i| = 1$ then, we set $\mathcal{O}(\sigma)$ to be the tree $\mathcal{O}(\sigma \backslash n)$ in which we add a new new binary node with leaves that replaces the leftmost leaf of $\mathcal{O}(\sigma \backslash n)$.*
  - *Else, let $k = c_{i,j-1}$, if k is the maximum element of $\mathcal{O}(\sigma \backslash n)$ then we add a new leaf to the last internal node of $\mathcal{O}(\sigma \backslash n)$ and else, we set $\mathcal{O}(\sigma)$ to be the tree $\mathcal{O}(\sigma \backslash n)$ in which a new binary node labelled nu with the smallest integer that does not appear as a label in $\mathcal{O}(\sigma \backslash n)$ and attach two new leaves to this internal node. Insert this binary node in $\mathcal{O}(\sigma \backslash n)$ by placing ν in the $k+1$-th leaf (we assume, for example, that the leaves are ordered in the depth-first order) of $\mathcal{O}(\sigma \backslash n)$.*

An example of the mapping $\mathcal{O}$ is depicted in Figure 4.6 on the same tree that was used in the other two bijections to illustrate the differences.

Theorem 4.2.11. *The map $\mathcal{O}$ is a one-to-one correspondence between $\mathcal{PR}$ and $\mathcal{T}$.*

Proof. The proof is very similar to the one in Theorem 4.2.4 and Theorem 4.2.7. At each step in the algorithm exactly one leaf is added to the tree (the total size is increased by 1). The corresponding behaviour on the tree for the last is unambiguous since it corresponds to a construction as in Proposition 4.2.3. Therefore the mapping $\mathcal{O}$ is also injective. □

$$\mathcal{T} \overset{\mathcal{O}}{\cong} \mathcal{PR}$$

From this bijection we get a very interesting between cycles in permutations and the depth of the leftmost leaf of the tree.

Theorem 4.2.12. *Let t be an increasing Schröder tree with n leaves and k is the depth of the leftmost leaf, and let $p = \mathcal{O}^{-1}(t)$. Let i be the number of cycles of p, then*

$$k = i - 1.$$

Proof. In the mapping $\mathcal{O}$ we see that when adding a new element in the permutation. We increase the depth of the leftmost leaf when a new cycle is created. The trivial tree (a binary

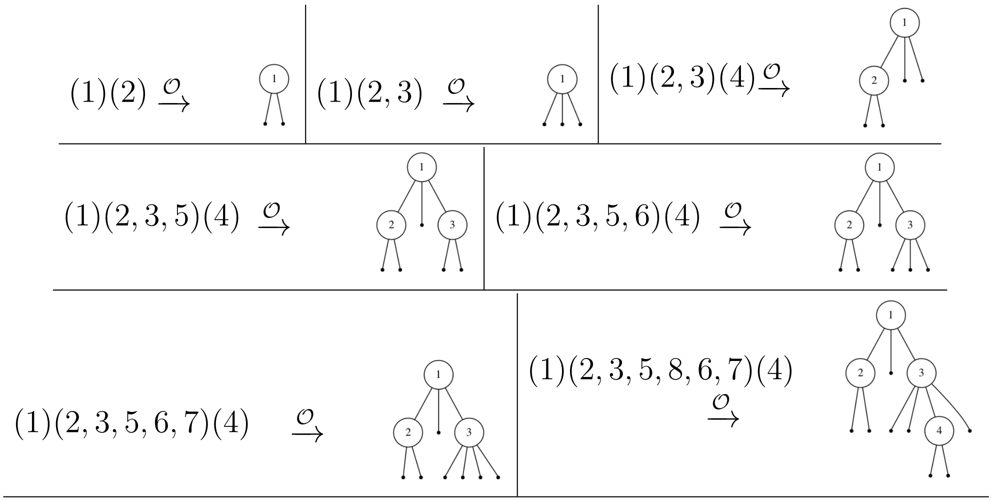

$(1)(2) \xrightarrow{\mathcal{O}}$
$(1)(2,3) \xrightarrow{\mathcal{O}}$
$(1)(2,3)(4) \xrightarrow{\mathcal{O}}$
$(1)(2,3,5)(4) \xrightarrow{\mathcal{O}}$
$(1)(2,3,5,6)(4) \xrightarrow{\mathcal{O}}$
$(1)(2,3,5,6,7)(4) \xrightarrow{\mathcal{O}}$
$(1)(2,3,5,8,6,7)(4) \xrightarrow{\mathcal{O}}$
1
2
3
4

Figure 4.6: A size-8 example of the mapping $\mathcal{O}$

root labelled 1) has depth of leftmost leaf 1. Its corresponding permutation has 2 elements and 2 cycles.

From these observations the result follows. □

Corollary 4.2.13. *Let $d_{n,k}$ be the number of increasing Schröder trees of size n such that the depth of the leftmost leaf is k. Let $PR_{n,i}$ be the number of n-permutations such that the elements 1 and 2 belong to different cycles and which have i cycles. Then,*

$$d_{n,k} = PR_{n,k+1}.$$

Proof. The result follows directly from Theorem 4.2.12. □

Corollary 4.2.14. *Let $t_{n,k}$ be the number of trees of size n with k internal nodes and let $d_{n,i}$ be the number of trees of size n where the leftmost leaf has depth i. Then,*

$$t_{n,k} = d_{n,n-k}.$$

Proof. The result is direct from Corollary 4.2.9 and Corollary 4.2.13. □

4.2.6 Relationship to Eulerian numbers and runs in permutations

Let us take the class of permutations $\mathcal{HP}$ defined in Section 4.2.5.1 which are permutations such that 1 appears before 2. We can make a partition of this set according to the number of runs. Let $q_{n,k}$ be the number of n-permutations belonging to $\mathcal{HP}$ that have k ***runs*** (see Section 3.1.2 for more details about runs). The first values of $q_{n,k}$ can be seen in Table 4.3 which is a shifted version of EIS A144696. It is easy to find a recurrence for $q_{n,k}$, for $n \geq 3$ and $2 \leq k \leq n$,

$$q_{n,k} = \begin{cases} 0 & \text{if } n = 2 \text{ and } k = 2, \\ 1 & \text{if } n \geq 1 \text{ and } k = 1, \\ k\, q_{n-1,k} \;+\; (n-k+1)\, q_{n-1,k-1} & n \geq 3 \text{ and } 2 \leq k \leq n \end{cases} \tag{4.5}$$

The last recurrence can be proved easily. There is exactly one permutation of each size that has one run which is $(1, 2, \dots, n)$. For size 2 there is only one permutation that counts, namely $(1, 2)$, since $(2, 1)$ is does not belong to $\mathcal{HP}$. Finally, the number of n-permutations having k runs either come from an $(n-1)$-permutation with k runs, in this case the last element can be added at the end of one of its runs so there are k places, or it comes from an $n-1$-permutation that has $k-1$ runs, in which case we want to create a new run and thus add is some other place than the $k-1$ places that do not create a new run. In this case there are n new places for the new element minus $k-1$ places that do not create a new run which give $n-(k-1) = n-k+1$.

1						
1	0					
1	2	0				
1	7	4	0			
1	18	33	8	0		
1	41	171	131	16	0	
1	88	718	1208	473	32	0

Table 4.3: The first values of $q_{n,k}$ with $n \in \{1, \dots, 7\}$ and $k \in \{1, \dots, n\}$

We define the *mirror permutation* of a permutation (written in the standard notation not the cycle notation) to be the permutation obtained by reading the permutation from right to left. We denote this operation by $mir(.)$. For instance, if $\sigma = (2, 3, 1)$ then $mir(\sigma) = (1, 3, 2)$.

Now if we take the mirror permutation of each permutation in $\mathcal{HP}$ we get all permutations and moreover,

Proposition 4.2.15. *Let σ be an n-permutation and $\sigma \in \mathcal{HP}$, if σ has k then $mir(\sigma)$ has $n-k+1$ runs.*

Proof. By reading the permutation from right to left, the sequence of ascents and descents is inversed. Therefore, the number of descents in σ was $k-1$, then $mir(\sigma)$ has $k-1$ ascents and $n-1-(k-1) = n-k$ descents and therefore $n-k+1$ runs. □

From Proposition 4.2.15, we see that it is possible to complete the values of $q_{n,k}$ by adding for each permutation in $\mathcal{HP}$ its mirror and with the partition on the number of *runs* we get *Eulerian numbers* that were presented in Section 3.1.2 and Table 3.2.

On the level of trees, there is a simple interpretation of ***runs*** in permutations of $\mathcal{HP}$ and the numbers $q_{n,k}$.

Proposition 4.2.16. *Let t be an increasing Schröder tree, m be the total number of internal nodes and k be the number of internal nodes (excepting the root node) that have an internal node attached to their second child. Let $\mathcal{M}(t)$, be the corresponding permutation in $\mathcal{HP}$. If $\mathcal{M}(t)$ has i runs, then,*

$$i = m - k.$$

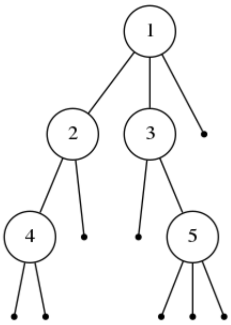


Figure 4.7: A tree of size 8 that has 5 internal nodes, one of which has a internal node as a second child (node labelled 3), the root node does not count

Proof. Through the mapping $\mathcal{M}$ when we add new element to the resulting permutation, if the element is added to the last position in the permutation we increase the arity of the last internal node by one. No new run is created and no new internal node is added.

Otherwise, when a new binary node b is added to the tree. An integer n is added in some position of the permutation (not the last one). We call this position j. In the next step the integer $n + 1$ will be added in the permutation. We know that if it is added to the right of n it will not create a new run. Since we have n in position j and a larger element at position $j + 1$. On the tree level this corresponds to adding a new binary node to the second child of b.

The argument still hold for separate steps. That is, if after adding b, in some future step we add an element l to the right of n. Two cases can arise, either the element to the right of n was smaller or larger than n. If it is smaller than n then, no new run is created.

However, if the integer l to the right of n is larger than n, then it has been added at some step in between the appearance of b and the actual step. A new run is created. But then l has already created a new binary node and the fresh binary node to be added in the tree will be added as a first position child of the node created by adding l.

Therefore, new nodes add runs in the permutation as long as they are not second child of some old node. □

Corollary 4.2.17. *If we let $t_{n,k}$ be the number of Increasing Schröder trees of size n, and are such that the difference between their number of internal nodes and the number of internal nodes (excepting the root node) that have another internal node attached to their second child is equal to k, then,*

$$t_{n,k} = q_{n,k}.$$

4.2.7 Analysis of typical parameters

In this section, our aim is to describe the *shape* of a *typical* increasing Schröder tree, i.e. a tree taken uniformly at random among all increasing Schröder tree of a fixed size. To get information about this *shape*, we focus on four characteristics of the tree: the number of internal nodes, the arity of the root, the number of leaves that are children of the root, and the number of binary nodes (node of arity 2). We show asymptotic theorems for these characteristics in a typical increasing Schröder tree when the size goes to infinity.

4.2.7.1 Quantitative analysis of the number of iteration steps

In this section, we show that although an increasing Schröder tree of size n can have between 1 and $n-1$ internal nodes, it typically has of order $n - \ln n$ internal nodes. This result is particularly interesting to analyse the complexity of the evolutionary process: this means that, on average, this evolutionary process takes of order $n - \ln n$ iteration steps to generate a typical increasing Schröder tree of size n. In fact, our result is stronger than just finding an equivalent for the average number of iterations since we prove a central limit theorem for this quantity. To complete the picture we also quantify the average number of nodes of a fixed degree. We will show that the average number of binary nodes in a typical tree is $n - 2\ln n$, the number of ternary nodes is $\ln n$ and higher arity nodes have a constant mean.

Theorem 4.2.18. *For all $n \geq 1$, we denote by X_n the number of internal nodes in a tree taken uniformly at random among all increasing Schröder trees of size n. Then, asymptotically when n tends to infinity, $\mathbb{E}_{\mathcal{T}_n}[X_n] \sim n - \ln n$, $\mathbb{V}_{\mathcal{T}_n}[X_n] \sim \ln n$, and*

$$\frac{X_n - (n - \ln n)}{\sqrt{\ln n}} \xrightarrow{d} \mathcal{N}(0,1) \quad \textit{in distribution.}$$

To prove this theorem, we enrich the specification Equation (4.2) with an additional parameter $\mathcal{U}$ marking the internal nodes:

$$\mathcal{T} = \mathcal{Z} \cup \left(\mathcal{U} \times \Theta_{\mathcal{Z}}\mathcal{T} \times \mathrm{Seq}_{\geq 1} \mathcal{Z}\right),$$

where the operator $\Theta_{\mathcal{Z}}$ consists in pointing an element marked by $\mathcal{Z}$. Remark here we do not use the Greene operator: the increasing labelling is a consequence of our point of view, we do not need to care about it. Using the symbolic method, this implies that, if $t_{n,k}$ is the number of increasing Schröder trees with n leaves and k internal nodes, $t_n(u) = \sum_{k=0}^{n-1} t_{n,k}\, u^k$, and $T(z,u) = \sum_{n\geq 1} t_n(u)\, z^n$, then

$$T(z,u) = z + \frac{uz^2}{1-z}\, \partial_z T(z,u), \tag{4.6}$$

where ∂_z denotes the partial differentiation according to z. Once again, we write $(1-z)T(z,u) = z(1-z) + uz^2$, and extract the coefficient of z^n on both sides; let us denoted by $t_n(u) = \sum_{k=0}^{n-1} t_{n,k}\, u^k$, then this gives $t_1(u) = 1$, $t_2(u) = u$ and, for all $n > 2$,

$$t_n(u) = (1 + (n-1)u)\, t_{n-1}(u). \tag{4.7}$$

Extracting the coefficient of u^k on both sides of this last equation gives: $t_{1,0} = 1$, $t_{n,1} = 1$ for all $n > 1$,

$$t_{n,k} = t_{n-1,k} + (n-1)\, t_{n-1,k-1} \qquad \text{for all } 0 < k < n,$$

and $t_{n,k} = 0$ otherwise. The first values of $t_{n,k}$ are listed in Table 4.4. Note that, for all $n \geq 1$, $t_{n,n-1}$ is the number of increasing binary trees (see [**FS09**, page 143] for details).

1						
0	1					
0	1	2				
0	1	5	6			
0	1	9	26	24		
0	1	14	71	154	120	
0	1	20	155	580	1044	720

Table 4.4: Values of $t_{n,k}$ (the number of increasing Schröder trees with n leaves and k internal nodes) for $n \in \{1, 2, \dots, 7\}$, and $k \in \{0, 1, \dots, n-1\}$.

From Equation (4.7), we easily deduce a closed form for $t_n(u)$: for all $n \geq 2$, we have

$$t_n(u) = u \prod_{\ell=2}^{n-1} (1 + \ell u). \tag{4.8}$$

This is a shifted version of the sequence EIS A145324, which is related to Stirling cycle numbers. Our proof of Theorem 4.2.18 relies on the following lemma, which is a straightforward consequence of Equation (4.8).

Lemma 4.2.19. *Let $SC_n(u) = \prod_{i=0}^{n-1}(u + i)$ be the generating functions of the respective rows of the Stirling Cycle numbers (see [**FS09**, page 735]), which enumerate all permutations of size n that decompose into k cycles (i.e. Stirling numbers of the first kind). If we set $\hat{t}_n(u) = \sum_{k=1}^{n} t_{n,k}\, u^{n-k}$, which is the row-reversed generating function, then*

$$\hat{t}_n(u) = \frac{SC_n(u)}{1 + u} = u \prod_{\ell=2}^{n-1} (u + \ell).$$

The row-reversed numbers appear in EIS A136124 and EIS A143491.

From Lemma 4.2.19 we can get an explicit formula for the numbers $\hat{t}_{n,k}$

Theorem 4.2.20. *Let $d_{n,k}$ be the number of trees with n leaves and depth of the leftmost leaf k. Let $t_{n,k}$ be the number of trees of size n with k internal nodes. Then,*

$$d_{n,k} = t_{n,n-k} = \hat{t}_{n,k} = \sum_{i=0}^{k} (-1)^{k-i} \begin{bmatrix} n \\ i \end{bmatrix}.$$

Proof. In Section 4.2.5.3, and corollary Corollary 4.2.14, we proved the relationship between the number of internal nodes and the depth of the leftmost leaf. The conclusion comes directly from Lemma 4.2.19 by extracting coefficients. □

Proof of Theorem 4.2.18. One could apply Hwang's quasi-powers theorem [**Hwa98**], but since we have an explicit formula for $t_n(u)$, we decide instead to apply Lévy's continuity theorem directly. By Lemma 4.2.19, we have that, if $\bar{X}_n = n - X_n$, for all $\xi \in \mathbb{R}$,

$$\mathbb{E}\left[\mathrm{e}^{i\xi\cdot\frac{\bar{X}_n-\ln n}{\sqrt{\ln n}}}\right] = \frac{1}{t_n}\,\mathrm{e}^{-i\xi\sqrt{\ln n}}\hat{t}_n\left(\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right) = \frac{2}{n!}\,\mathrm{e}^{-i\xi\sqrt{\ln n}+\frac{i\xi}{\sqrt{\ln n}}}\cdot\frac{\Gamma\left(n+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right)}{\Gamma\left(2+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right)}$$

$$= \frac{2+o(1)}{\Gamma(3+o(1))}\,\frac{\left(n-1+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right)^{n+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}-\frac{1}{2}}\mathrm{e}^{n}\mathrm{e}^{-i\xi\sqrt{\ln n}}}{\mathrm{e}^{n-1+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}}\,n^{n+\frac{1}{2}}},$$

where we have used Stirling's formula. Note that

$$\lim_{n\to\infty}\mathrm{e}^{1-\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}} = 1,$$

and $\Gamma(3) = 2$, which implies that

$$\begin{aligned}\mathbb{E}\left[\mathrm{e}^{i\xi\cdot\frac{\bar{X}_n-\ln n}{\sqrt{\ln n}}}\right] &= (1+o(1))\frac{\left(n-1+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right)^{n+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}-\frac{1}{2}}\mathrm{e}^{-i\xi\sqrt{\ln n}}}{n^{n+1/2}}\\ &= (1+o(1))\,\frac{n^{\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}}}{n}\left(1+O\left(\frac{1}{n\sqrt{\ln n}}\right)\right)^{n-\frac{1}{2}+\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}}\mathrm{e}^{-i\xi\sqrt{\ln n}}\\ &= (1+o(1))\,\frac{n^{\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}}\mathrm{e}^{-i\xi\sqrt{\ln n}}}{n}.\end{aligned}$$

Since

$$\begin{aligned}n^{\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}} &= \exp\left((\ln n)\mathrm{e}^{\frac{i\xi}{\sqrt{\ln n}}}\right)\\ &= \exp\left((\ln n)\left(1+\frac{i\xi}{\sqrt{\ln n}}-\frac{\xi^2}{2\ln n}+O\left((\ln n)^{-3/2}\right)\right)\right)\\ &\underset{n\to\infty}{=} n\mathrm{e}^{i\xi\sqrt{\ln n}-\xi^2/2},\end{aligned}$$

we get

$$\mathbb{E}\left[\mathrm{e}^{i\xi\cdot\frac{\bar{X}_n-\ln n}{\sqrt{\ln n}}}\right] = (1+o(1))\,\mathrm{e}^{-\xi^2/2},$$

which, by Lévy's continuity theorem concludes the proof; recall that $\bar{X}_n = n - X_n$. □

4.2.7.2 Quantitative characteristics of the root node

In this section, we study two parameters of the root of a typical increasing Schröder tree: the total number of its children (i.e. its arity), and the number of its children that are leaves.

We denote by $A_n^{\mathcal{T}}$ the arity of the root in a tree picked uniformly at random among all *increasingSchröder* trees of size n, and by p_n its probability generating function:

$$p_n(u) = \sum_{k\geq 0}\mathbb{P}(A_n^{\mathcal{T}} = k)\,u^k.$$

Theorem 4.2.21. *The degree of the root of a tree taken uniformly at random among all increasing Schröder trees of size n when n tends to infinity has,*

$$\mathbb{E}[A_n] \underset{n\to\infty}{\sim} 2\,\mathrm{e}-3, \quad \textit{and} \quad \mathbb{V}[A_n] \underset{n\to\infty}{\sim} 14\,\mathrm{e}-4\,\mathrm{e}^2-8.$$

The limiting law is a modified Poisson law. That is

$$p_n(u) \underset{n\to\infty}{\sim} \left(2-2\,u^{-1}\right)\mathrm{e}^u-u+2\,u^{-1}.$$

And the second result is:

Theorem 4.2.22. *Let L_n be the number of children of the root that are leaves in a tree taken uniformly at random among all increasing Schröder trees of size n. Asymptotically when n tends to infinity,*

$$\mathbb{E}[L_n] = \frac{2\mathrm{e}}{n} + \Theta\left({}^1\!/\!_{(n\cdot n!)}\right) \quad \textit{and} \quad \mathbb{V}[L_n] = \frac{2\mathrm{e}}{n} + \Theta\left({}^1\!/\!_{n^2}\right).$$

Theorem 4.2.21 is a direct consequence of the following lemma.

Lemma 4.2.23. *If $t_{n,k}$ is the number of increasing Schröder trees whose root has arity k, then $t_{1,0}=1$, for all $n \geq 0, t_{n,1}=0$ and for all $n \geq 2, 2 \leq k \leq n-1$,*

$$t_{n,k} = \frac{k\;n!}{(k+1)!}, \quad \textit{and } t_{n,n}=1.$$

Indeed, this lemma together with the fact that $t_n = {}^{n!}\!/\!_2$, imply, for all $2 \leq k < n$,

$$\mathbb{P}(A_n=k) = \frac{2k}{(k+1)!}.$$

From here, the mean and variance can be mechanically computed and we have as n tends to infinity

$$\begin{aligned}
p_n(u) &\underset{n\to\infty}{\sim} \sum_{k=2}^{\infty} \frac{2\,k}{(k+1)!}u^k \\
&= \sum_{k=2}^{\infty} \frac{(2\,(k+1)-2)}{(k+1)!}u^k \\
&= 2\,\left(\mathrm{e}^u-1-u\right) - 2\sum_{k=2}^{\infty}\frac{1}{(k+1)!}u^k \\
&= 2\,\left(\mathrm{e}^u-1-u\right) - \frac{2}{u}\sum_{k=2}^{\infty}\frac{u^{k+1}}{(k+1)!} \\
&= 2\,\left(\mathrm{e}^u-1-u\right) - \frac{2}{u}\left(\mathrm{e}^u-1-u-\frac{u^2}{2}\right) \\
&= \left(2-2\,u^{-1}\right)\mathrm{e}^u-u+2\,u^{-1}
\end{aligned}$$

which concludes the proof of Theorem 4.2.21.

1,	0						
0,	0,	1					
0,	0,	2,	1				
0,	0,	8,	3,	1			
0,	0,	40,	15,	4,	1		
0,	0,	240,	90,	24,	5,	1	
0,	0,	1680,	630,	168,	35,	6,	1

Table 4.5: Values of $t_{n,k}$, the number of size-n increasing Schröder trees of root-arity k, and $0 \leq k \leq n \in \{1, \dots, 7\}$.

We refer the reader to Table 4.5 where the first values of $t_{n,k}$ are listed. The sequences $(t_n(u))_{n\geq 1}$ and $(t_{n,k})_{2\leq k<n}$ are related to the sequences EIS A094112 and EIS A092582, which enumerate some families of permutations (the former enumerates a family of permutations avoiding some pattern, the second permutations with initial cycle of a given size). Since the number of increasing Schröder trees of size $n \geq 2$ is equal to $^{n!}/_{2}$, it is natural to expect some links between this family of trees and permutations that we have exhibited in Section 4.2.5.

Proof of Lemma 4.2.23. In this proof, the variable $\mathcal{U}$ marks the arity of the root (we re-use the same notation as in the previous section, but with a different meaning; this is done to avoid having too many different notations). Using the evolution process, we get that

$$\mathcal{T} = \mathcal{Z} \cup \big(\mathcal{U} \times \mathcal{Z} \times \mathrm{Seq}_{\geq 1}(\mathcal{U} \times \mathcal{Z})\big) \cup \big(\Theta_{\mathcal{Z}}(\mathcal{T} \setminus \mathcal{Z}) \times \mathrm{Seq}_{\geq 1} \mathcal{Z}\big)\,.$$

Indeed, the root is either a leaf ($\mathcal{Z}$), or it is an internal node to which is attached a sequence of at least 2 leaves ($\mathcal{U} \times \mathcal{Z} \times \Theta_{\mathcal{Z}}(\mathcal{Z}) \times \mathrm{Seq}_{\geq 1}(\mathcal{U} \times \mathcal{Z})$), or the tree is larger, i.e. the last step in the evolution process was replacing another leaf by an internal node to which is attached a sequence of non-marked leaves ($\Theta_{\mathcal{Z}}(\mathcal{T} \setminus \mathcal{Z}) \times \mathrm{Seq}_{\geq 1} \mathcal{Z}$). Using the symbolic method, we thus get that

$$T(z, u) = z + \frac{u^2 z^2}{1 - uz} + \frac{z^2}{1 - z}\, \partial_z \left(T(z, u) - z\right).$$

In the same way as before, through a direct extraction $[z^n](1 - zu)(1 - z)T(z, u)$, we prove that $t_1(u) = 1$, $t_2(u) = u^2$, and for all $n > 2$,

$$t_n(u) = (u - 1)\, u^{n-1} + n\, t_{n-1}(u).$$

This implies $t_{1,0} = 1$, $t_{n,n} = 1$ for all $n \geq 1$, $t_{n,k} = n\, t_{n-1,k}$ for all $1 \leq k \leq n - 1$, and $t_{n,k} = 0$ for all $k > n$, which concludes the proof. □

Proof of Theorem 4.2.22. The operators needed for the specification are not so classical so we prefer to directly write the differential equation satisfied by $T(z, u) = \sum_{n,k} t_{n,k}\, u^k z^n$, where $t_{n,k}$ is the number of size-n increasing Schröder trees with k leaves attached to the root. Like in the proof of Lemma 4.2.23 at each step we must remove the tree reduced to the leaf, i.e. $\mathcal{T} \setminus \mathcal{Z}$. So let us introduce the function $V(z, y) = T(z, y) - z$. Thus we get

$$T(z, u) = z + \frac{u^2 z^2}{1 - uz} + \frac{z^2}{1 - z}\, \frac{\partial_u V(z, u)}{z} + \frac{z^2}{1 - z} \Big(\partial_z V(z, u) - \frac{u}{z}\, \partial_u V(z, u)\Big)\,. \quad (4.9)$$

Indeed, by looking at the last step in the evolution process, four possibilities occur:
- the tree is reduced to a leaf z, i.e. the evolution process did not already start
- the tree contains a single internal node to which a sequence of at least 2 leaves is attached ($\frac{u^2\,z^2}{1-uz}$), i.e. the evolution process has gone through one step only,
- in the last step of the evolution process, a leaf of the root has been replaced by an internal node to which a sequence of leaves is attached ($\frac{z^2}{1-z}\,\frac{\partial_u V(z,u)}{z}$), in fact, leaves attached to the root are marked as zu, the differentiation by u followed by the division by z gives the result,
- in the last step of the evolution process, a leaf that is not attached to the root has been selected and replaced by an internal node attached to at least two leaves:

$$\frac{z^2}{1-z}\,\left(\partial_z V\left(z,u\right) - \frac{u}{z}\,\partial_u V\left(z,u\right)\right).$$

The second term removes the trees built in the first one where a leaf attached to the root has been selected. As an example, take a tree counted by $z^{\ell+r}u^r$, thus containing ℓ leaves such that r of them are attached to the root. The operation gives $(\ell+r)z^{\ell+r-1}u^r - \frac{u}{z}\,r\,z^{\ell+r}u^{r-1}$ and thus gives exactly $\ell z^{\ell+r-1}u^r$.

After some simplifications and multiplications by $(1-uz)(1-z)$ we get

$$(1-uz)(1-z)\;V(z,u) = u^2z^2(1-z) + z^2(1-uz)\left(\partial_z V\left(z,u\right) - \frac{u}{z}\,\partial_u V\left(z,u\right)\right).$$

By extracting the coefficient of z^n from the latter equation, we directly get that, for all $n \geq 4$,

$$v_n(u) = (n+u)\;v_{n-1}(u) - u(n-1)\;v_{n-2}(u) - (u-1)\;v'_{n-1}(u) + u(u-1)\;t'_{n-2}(u),$$

and $v_1(u) = 0$, $v_2(u) = u^2$ and $v_3(u) = 2u^2 + u^3$.

To evaluate the average number of leaves attached to the root we must compute the limit of the ratio $v'_n(u)/v_n(u)$ when n tends to infinity and evaluate it for $u = 1$. Differentiating the last equation we get

$$\begin{aligned} v'_n(u) \;\; &= \;\; v_{n-1}(u) + (n+u-1)\;v'_{n-1}(u) - (u-1)\;v''_{n-1}(u) - (n-1)\;v_{n-2}(u) \\ &\quad +(2u-(n-1)u-1)\;v'_{n-2}(u) + u(u-1)\;v''_{n-2}(u). \qquad (4.10) \end{aligned}$$

We thus define the sequence of the average values $m_n = v'_n(1)/v_n(1)$ and get for $n \geq 4$

$$m_n = m_{n-1} - \frac{n-2}{n(n-1)}m_{n-2},$$

with $m_1 = 0, m_2 = 2$ and $m_3 = 5/3$. In order to analyse the sequence of real values m_n we introduce an alternative sequence ℓ_n such that $\ell_n = n\;m_n$ and thus we obtain for all $n \geq 1$, we get, for all $n \geq 4$,

$$\ell_n = \left(1 + \frac{1}{n-1}\right)\ell_{n-1} - \frac{1}{n-1}\ell_{n-2}, \qquad (4.11)$$

and $\ell_1 = 0, \ell_2 = 4$ and $\ell_3 = 5$. Finally, we set $e_n = 2\sum_{i=0}^{n-1} 1/i!$ for all $n \geq 1$, and prove by induction that, for all $n \geq 1$, $\ell_n = e_n$. First note that $\ell_n = e_n$ for $n = \{1,2,3\}$. Now take $n \geq 4$ and assume that for all $i < n$ we have $\ell_i = e_i$. Using the fact that $e_n = e_{n-1} + 2/(n-1)!$, and Equation Equation (4.11), we have

$$\ell_n - e_n = \ell_{n-1} + \frac{1}{n-1}\left(\ell_{n-1} - \ell_{n-2}\right) - e_n$$

$$= \ell_{n-1} - e_{n-1} + \frac{1}{n-1}\left(\ell_{n-1} - \ell_{n-2} - \frac{2}{(n-2)!}\right)$$
$$= \ell_{n-1} - e_{n-1} + \frac{1}{n-1}\left(\ell_{n-1} - e_{n-1} - (\ell_{n-2} - e_{n-2})\right) = 0,$$

and thus $\ell_n = e_n$, which concludes the induction argument. Since, by definition of e_n, $e_n = 2\mathrm{e} + \Theta({}^1\!/_{n!})$, and since $e_n = \ell_n = nm_n = n\mathbb{E}[L_n]$, we get

$$m_n = \mathbb{E}[L_n] = \frac{2}{n}\sum_{i=0}^{n-1}\frac{1}{i!} \underset{n\to\infty}{=} \frac{2\mathrm{e}}{n} + \Theta\left({}^1\!/_{(n\cdot n!)}\right) \tag{4.12}$$

We now estimate the variance of L_n; to do so, we use the following identity (see, e.g., [FS09, p. 159]):

$$\mathbb{V}[L_n] = \mathbb{E}[L_n(L_n-1)] + \mathbb{E}[L_n] - (\mathbb{E}[L_n])^2 \tag{4.13}$$

Since we already have estimated $\mathbb{E}L_n$, we only need to estimate $\mathbb{E}[L_n(L_n-1)] = v_n''(1)/v_n(1)$, which we denote by k_n. Differentiating Section 4.2.7.2 we get that, for all $n \geq 4$,

$$k_n = k_{n-1} - \frac{1}{n}\left(k_{n-1} - \frac{n-3}{n-1}k_{n-2}\right) + \frac{2}{n}\left(m_{n-1} - \frac{n-2}{n-1}m_{n-2}\right),$$

where we recall that $m_n = v_n'(1)/v_n(1) = \mathbb{E}[L_n]$. The first terms of $(k_n)_{n\geq 1}$ are $k_1 = 0, k_2 = 2$ and $k_3 = 2$. For all $n \geq 1$, set $r_n = n(n-1)k_n$. Using Equation (4.12), we get that, for all $n \geq 4$,

$$r_n = r_{n-1} + \frac{1}{n-2}(r_{n-1} - r_{n-2}) + \frac{4}{(n-2)!},$$

with the initial values $r_1 = 0, r_2 = 4$ and $r_3 = 12$. Finally, for all $n \geq 3$, we set

$$\tilde{e}_n = 4\sum_{i=2}^{n-1}\frac{i-1}{(i-2)!} = 4\sum_{i=3}^{n-1}\frac{1}{(i-3)!} + 4\sum_{i=2}^{n-1}\frac{1}{(i-2)!},$$

and $\tilde{e}_1 = 0$, $\tilde{e}_2 = 4$. By induction, one can prove that, for all $n \geq 1$, $r_n = \tilde{e}_n$, which implies

$$k_n = \mathbb{E}[L_n(L_n-1)] = \frac{r_n}{n(n-1)} = \frac{4}{n(n-1)}\sum_{i=3}^{n-1}\frac{1}{(i-3)!} + \frac{4}{n(n-1)}\sum_{i=2}^{n-1}\frac{1}{(i-2)!}$$
$$= \frac{8\mathrm{e}}{n^2} + \Theta\left({}^1\!/_{n^3}\right).$$

Using this last estimate together with Equation (4.13) Equation (4.12), we get

$$\mathbb{V}L_n \underset{n\to\infty}{=} \frac{2\mathrm{e}}{n} + \Theta\left({}^1\!/_{n^2}\right). \qquad \square$$

4.2.7.3 Quantitative analysis of the number of nodes of a given arity

In this section, we prove asymptotic results for the number of nodes of a given arity in a typical increasing Schröder tree, starting with binary nodes:

Theorem 4.2.24. *Let B_n be the number of binary nodes (nodes of arity 2) in a tree taken uniformly at random among all increasing Schröder trees of size n. Asymptotically when n*

tends to infinity, we have

$$\mathbb{E}[B_n] = n - 2\,\ln n + 2\gamma - \frac{7}{3} + \mathcal{O}(1/n), \qquad \textit{and}\ \mathbb{V}[B_n] = 4\,\ln n + 4\,\gamma - \frac{2}{3}\pi^2 - \frac{17}{6} + \mathcal{O}(1/n),$$

where γ is the Euler-Mascheroni constant. Moreover, in distribution when $n \to +\infty$,

$$\frac{B_n - (n - 2\ln n)}{2\sqrt{\ln n}} \to \mathcal{N}(0,1).$$

In other words, almost all internal nodes are binary, only a proportion of order $2\ln n/n$ of internal nodes are at least ternary. In the following theorem, we show that, on average, half of all non-binary nodes are ternary.

Theorem 4.2.25. *Let $C_n^{(\ell)}$ be the number of nodes of arity $\ell \geq 3$ in a tree taken uniformly at random among all increasing Schröder trees of size n. Asymptotically when n tends to infinity, we have*

$$\mathbb{E}C_n^{(3)} = \ln n + \mathcal{O}(1), \qquad \textit{and}\ \mathbb{E}C_n^{(4)} \sim c_\ell,$$

for some positive constants $(c_\ell)_{\ell\geq 4}$; and, for $\ell = 4$, we have $c_4 = 23/90$.

Proof of Theorem 4.2.24. Here the specification is easy to exhibit, and its translation via the symbolic method is direct (in this proof, $\mathcal{U}$ marks the binary nodes):

$$\mathcal{T} = \mathcal{Z} \cup \left(\Theta_{\mathcal{Z}}\mathcal{T} \times \left(\mathcal{U} \times \mathcal{Z} \cup \mathrm{Seq}_{\geq 2}\,\mathcal{Z}\right)\right);$$

$$T(z,u) = z + \left(u\,z^2 + \frac{z^3}{1-z}\right)\partial_z T(z,u).$$

The method we use to analyse this differential equation is similar to [**CHY00**]. For all $n \geq 3$,

$$t_n(u) = (1 + u(n-1))t_{n-1}(u) + (1-u)(n-2)t_{n-2}(u), \tag{4.14}$$

with $t_1(u) = 1$, $t_2(u) = u$, and $t_3(u) = 1 + u^2$. Once again (see also Lemma 4.2.19) we take the row-reversed generating function $\hat{t}_n(u) = \sum_{k=1}^n t_{n,k}\ u^{n-k} = t^n t_n(1/u)$. From Equation (4.14), we get that, for all $n \geq 4$,

$$\hat{t}_n(u) = \frac{n + u - 1}{n}\ \hat{t}_{n-1}(u) + \frac{u(u-1)(n-2)}{n(n-1)}\ \hat{t}_{n-2}(u), \tag{4.15}$$

with $\hat{t}_2(u) = u$ and $\hat{t}_3(u) = (2u + u^3)/3$. Let us now define $F(z,u) = \sum_{n\geq 2} n\hat{t}_n(u)z^n$; this generating function satisfies the following differential equation:

$$z\,(1-z)\,\partial_z F(z,u) = \left(1 + uz - u(1-u)z^2\right)\ F(z,u) + 2uz^2\,(1 - u(1-u)z)\,,$$

with initial condition $\partial_z^2 F(z,u)|_{z=0} = 4\,u$. This last equation gives

$$F(z,u) = 2uz\,\exp\left(u(1-u)z\right)(1-z)^{-1-u^2} \int_0^z (1 - u(1-u)t)\,(1-t)^{u^2} \exp\left(-u(1-u)t\right)\,\mathrm{d}t.$$

Let $\phi(z,u) = (1 - u(1-u)z)\,(1-z)^{u^2}\mathrm{e}^{-u(1-u)z}$; with this definition, we get

$$F(z,u) = (1-z)^{-1-u^2}\left(g(u) + E(z,u)\right),$$

where,

$$g(u) = 2u\mathrm{e}^{u(1-u)}\,(1-z)^{-1-u^2} \int_0^1 \phi(t,u)\,\mathrm{d}t$$

and,

$$E(z,u) = \left(z\,\mathrm{e}^{u(1-u)z}\right) - \mathrm{e}^{u(1-u)} \int_0^1 \phi(t,u)\,dt - z\mathrm{e}^{u(1-u)z} \int_z^1 \phi(t,u)\,\mathrm{d}t.$$

Therefore, asymptotically when $n \to +\infty$,

$$n\hat{t_n}(u) = \frac{g(u)}{\Gamma\left(1+u^2\right)}\, n^{u^2}\left(1 + \mathcal{O}(1/n)\right)$$

uniformly for all u such that $|u-1| \le \delta$ for some $\delta > 0$. This thus falls into the scope of the quasi-powers framework and Theorem IX.8 [**FS09**, p. 645] is applicable with $B(u) = \exp(2u)$ and $\beta_n = \ln n$, which concludes the proof (the mean and variance expansions can be calculated automatically using, e.g., a computer software such as Maple). □

Proof of Theorem 4.2.25. We first look at ternary nodes: the specification (with $\mathcal{U}$ marking ternary nodes) is given by

$$\mathcal{T} = \mathcal{Z} \cup \left(\Theta_{\mathcal{Z}}\mathcal{T} \times \left(\mathcal{Z} \cup \mathcal{U} \times \mathcal{Z}^2 \cup \operatorname{Seq}_{\ge 3} \mathcal{Z}\right)\right)$$

which implies

$$T(z,u) = z + \left(z^2 + u\,z^3 + \frac{z^4}{1-z}\right)\partial_z T(z,u),$$

and thus, for all $n \ge 4$:

$$t_n(u) = nt_{n-1}(u) + (u-1)(n-2)t_{n-2}(u) + (n-3)(1-u)t_{n-3}(u),$$

with $t_1(u) = 1$ and $t_2(u) = 1$. Differentiating this equation, we get that, for all $n \ge 5$,

$$t'_n(u)|_{u=1} = nt'_{n-1}(u)|_{u=1} + \frac{(n-2)(n-2)!}{2} - \frac{(n-3)(n-3)!}{2}.$$

This thus implies that, for all $n \ge 5$,

$$\mathbb{E}[C_n^{(3)}] = \mathbb{E}[C_{n-1}^{(3)}] + \frac{(n-2)}{n(n-1)} - \frac{(n-3)}{n(n-1)(n-2)} = \frac{10}{24} + \sum_{\ell=5}^{n}\left(\frac{(k-2)}{k(k-1)} - \frac{(k-3)}{k(k-1)(k-2)}\right).$$

since $\mathbb{E}[C_4^{(3)}] = 10/24$. Using again the fact that $\sum_{k=1}^n \frac{1}{k} = \ln n + \mathcal{O}(1)$ and $\sum_{k=1}^n \frac{1}{k^2} = \mathcal{O}(1)$ when n tends to infinity, we get

$$\mathbb{E}[C_n^{(3)}] = \ln n + \mathcal{O}(1),$$

as claimed.

We reason similarly for $\ell = 4$ ($\mathcal{U}$ now marks nodes of arity 4):

$$\mathcal{T} = \mathcal{Z} \cup \left(\Theta_{\mathcal{Z}}\mathcal{T} \times \left(\mathcal{Z} \cup \mathcal{Z}^2 \cup \mathcal{U} \times \mathcal{Z}^3 \cup \operatorname{Seq}_{\ge 4} \mathcal{Z}\right)\right);$$

$$T(z,u) = z + \left(z^2 + z^3 + u\,z^4 + \frac{z^5}{1-z}\right)\partial_z T(z,u).$$

Thus, for all $n \ge 4$, we have

$$t_n(u) = nt_{n-1}(u) + (u-1)(n-3)t_{n-3}(u) + (n-4)(1-u)t_{n-4}(u),$$

with $t_1(u) = 1$ and $t_2(u) = 1$, which, after differentiating at $u = 1$ and dividing by t_n gives

$$\mathbb{E}[C_n^{(4)}] = \mathbb{E}[C_{n-1}^{(4)}] + \frac{(n-3)}{n(n-1)(n-2)} - \frac{(n-4)}{n(n-1)(n-2)(n-3)},$$

with $\mathbb{E}[C_5^{(4)}] = {}^{12}\!/_{120}$. A simple look to this recurrence shows that it converges to a constant since it is a modified geometric sum. Solving the recurrence we obtain,

$$\mathbb{E}[C_n^{(4)}] = \frac{23}{90} - \frac{13}{6n} - \frac{1}{6\,(n-2)} + \frac{4}{3\,(n-1)},$$

which proves the statement for $\ell = 4$.

Let us now treat the general $\ell \geq 5$ case ($\mathcal{U}$ now marks nodes of arity ℓ):

$$\mathcal{T} = \mathcal{Z} \cup \left(\Theta_{\mathcal{Z}}\mathcal{T} \times \left(\left(\cup_{i=1}^{\ell-2} \mathcal{Z}^{\ell}\right) \cup \mathcal{U} \times \mathcal{Z}^{\ell-1} \cup \text{Seq}_{\geq 4}\,\mathcal{Z}\right)\right);$$

$$T(z,u) = z + \left(\left(\sum_{i=2}^{\ell-1} z^i\right) + u\, z^{\ell} + \frac{z^{\ell+1}}{1-z}\right) \partial_z T(z,u).$$

This implies that, for all $n \geq \ell$:

$$t_n(u) = n t_{n-1}(u) + (u-1)(n-\ell+1)t_{n-\ell+1}(u) + (n-\ell)(1-u)t_{n-\ell}(u),$$

with $t_n(u) = 1$ for all $n < \ell$. Therefore, we get

$$\begin{aligned}\mathbb{E}[C_n^{(\ell)}] &= \mathbb{E}[C_{n-1}^{(\ell)}] + \frac{(n-\ell+1)}{n(n-1)\cdots(n-\ell+2)} - \frac{(n-\ell)}{n(n-1)\cdots(n-\ell+1)} \\ &= \mathbb{E}[C_{\ell}^{(\ell)}] + \sum_{k=\ell+1}^{n} \left(\frac{(k-\ell+1)}{k(k-1)\cdots(k-\ell+2)} - \frac{(k-\ell)}{k(k-1)\cdots(k-\ell+1)}\right).\end{aligned}$$

Since

$$\left(\frac{(k-\ell+1)}{k(k-1)\cdots(k-\ell+2)} - \frac{(k-\ell)}{k(k-1)\cdots(k-\ell+1)}\right) \underset{k\to\infty}{\sim} \frac{1}{k^{\ell-2}},$$

which implies that, for all $\ell \geq 4$,

$$\lim_{n\to\infty} \mathbb{E}[C_n^{(\ell)}] = \mathbb{E}[C_{\ell}^{(\ell)}] + \sum_{k=\ell+1}^{\infty} \left(\frac{(k-\ell+1)}{k\cdots(k-\ell+2)} - \frac{(k-\ell)}{k\cdots(k-\ell+1)}\right) < +\infty.$$

All these recurrences converges to constants that get smaller and smaller when ℓ increases. □

Note that the constants c_{ℓ} are computable by solving the simple recurrences for each case; Figure 4.8 gives a summary of the typical number of nodes for the smallest arities.

	2-ary	3-ary	4-ary	5-ary	6-ary	7-ary	8-ary	9-ary	10-ary
$\mathbb{E}C_n^{(\ell)}$	$n - 2\ln n$	$\ln n$	$\frac{23}{90}$	$\frac{1}{32}$	$\frac{107}{25200}$	$\frac{47}{86400}$	$\frac{101}{1587600}$	$\frac{229}{33868800}$	$\frac{659}{1005903360}$

Figure 4.8: The asymptotic number of ℓ-ary nodes

4.2.7.4 Typical depth of the leftmost leaf

In this section, we prove a central limit theorem for the depth of the leftmost leaf in a typical increasing Schröder tree; this gives us a lower bound for the height of a typical increasing Schröder tree.

Lemma 4.2.26. *Let Y_n be the depth of the leftmost leaf in a tree taken uniformly at random among all increasing Schröder trees of size n. For all $n \geq 1$, $Y_n = n - X_n$, where X_n is the number of internal nodes in a typical increasing Schröder tree of size n (see Theorem 4.2.18), and thus, we have convergence in distribution when n tends to infinity:*

$$\frac{Y_n - \ln n}{\sqrt{\ln n}} \xrightarrow{d} \mathcal{N}(0,1).$$

Note that the choice of the leftmost leaf is arbitrary, although it has the advantage that the specification is straightforward. Table 4.6 exhibits the first values of $(t_{n,k})$.

Proof. We directly look at the differential equation satisfied by $T(z,u)$, where u marks the internal nodes that belong to the leftmost path (between the root and the leftmost leaf).

$$T(z,u) = z + \partial_z \left(\frac{T(z,u)}{z} \right) \frac{z^3}{1-z} + T(z,u) \frac{uz}{1-z}.$$

Indeed, the tree is either a unique leaf (which is thus also the leftmost leaf) at height zero (z), or at the last step of the evolution process, we have selected a leaf that is not the leftmost one and replaced it by a sequence of at least two leaves ($\partial_z(T(z,u)/z)\frac{z^3}{1-z}$), or we have replaced the leftmost leaf by an internal node and a sequence of at least two leaves ($T(z,u)\frac{uz}{1-z}$). We rewrite this equation as

$$(1-uz)T(z,u) = z(1-z) + z^2 \partial_z T(z,u),$$

and thus, identifying the coefficient of z^n on both sides gives that

$$t_n(u) = (u+n-1)t_{n-1}(u) \quad (\forall n \geq 3),$$

$t_1(u) = 1$, and $t_2(u) = u$. This implies that, for all $n \geq 3$,

$$t_n(u) = u \prod_{i=2}^{n-1} (u+i) = \frac{1}{1+u} SC_n(u),$$

where $SC_n(u)$, defined in Lemma 4.2.19, is the generating function of all size n permutations with k cycles. Therefore, using Lemma 4.2.19, we get that $Y_n = n - X_n$ in distribution, where X_n is the number of internal nodes in a typical increasing Schröder tree, which concludes the proof, by Theorem 4.2.18. □

1						
0	1					
0	2	1				
0	6	5	1			
0	24	26	9	1		
0	120	154	71	14	1	
0	720	1044	580	155	20	1

Table 4.6: The values of $t_{n,k}$, the number of increasing Schröder trees of size n trees whose leftmost leaf has depth k, for all $0 \leq k < n \in \{1, 2, \dots, 7\}$.

These numbers appear in EIS A136124 and EIS A143491.

As we have seen before these numbers are the row reversed of the triangle of internal nodes. We have exhibited these links in Section 4.2.5.2 and with Corollary 4.2.14.

4.2.8 Analysis of the height of a typical increasing Schröder tree

It is possible to define an incremental process (in the sense of Section 3.5.5) to generate uniformly randomly a tree of increasing Schröder tree using the recurrence Equation (4.4). Therefore we can build a tree using successive insertions in the tree to add exactly one leaf at each iteration step based on a fixed probability distribution.

This probabilistic construction used in our uniform sampler that is given in Chapter 7 and Algorithm 1 allows us to prove the following result.

Theorem 4.2.27. *For all $n \geq 2$, let H_n be the height of a tree taken uniformly at random among all increasing Schröder trees of size n. Asymptotically when n tends to infinity,*

$$\mathbb{P}\left(\frac{H_n}{\ln n} \in [1-\varepsilon, \gamma+\varepsilon]\right) \to 1,$$

where $\gamma = \inf\{c > 0 \colon c - 1 + c\ln(2/c) < 0\} \approx 4.311$. This implies, in particular that $\mathbb{E}[H_n] = \Theta(\ln n)$ when n tends to infinity.

Definition 4.2.28. *Given a sequence of integers $\boldsymbol{d} = (d_i)_{i\geq 1}$, we define the random $\boldsymbol{d}$-ary tree $(\tau_n^{(\boldsymbol{d})})_{n\geq 0}$ recursively as follows: $\tau_0^{(\boldsymbol{d})}$ is reduced to its root, given $\tau_{\ell-1}^{(\boldsymbol{d})}$, we build $\tau_\ell^{(\boldsymbol{d})}$ as the tree obtained by picking a leaf uniformly at random in $\tau_{\ell-1}^{(\boldsymbol{d})}$ and replacing it by a node to which d_ℓ leaves are attached.*

Lemma 4.2.29. *Let $\boldsymbol{D} = (D_\ell)_{\ell\geq 1}$ be the sequence of integer-valued random variables defined by:*

- $\mathbb{P}(D_1 = k) = 2k/(k+1)!$ *for all $k \geq 2$, and*
- *if, for all $\ell \geq 1$, we denote by $\bar{D}_\ell = \sum_{i=1}^{\ell} D_i$, then,*

$$\mathbb{P}(D_{\ell+1} = k | D_1, \dots, D_\ell) = \frac{(\bar{D}_\ell + 1)!(k - 1 + \bar{D}_\ell)}{(k + \bar{D}_\ell)!}.$$

Then, for all $\ell \geq 1$, the tree $\tau_\ell^{(\boldsymbol{D})}$ given its size is equal in distribution to an increasing Schröder tree taken uniformly at random among all trees of that size.

Proof. This follows from Theorem 7.2.1. Indeed, note that the degree of the last inserted internal node increases as long as the random integer $k = k_i$ (see line 7 of Algorithm 1) drawn in the i-th loop is not equal to i. Note that this happens with probability $1/i$. For example, the degree of the root starts at 2, we draw the first integer $k_3 \in \{1, 2, 3\}$ and if $k_3 \neq 3$, then we can conclude that $D_1 = 2$, otherwise, we know that $D_1 \geq 3$ and we need to look at k_4. Therefore, $\mathbb{P}(D_1 = 2) = 2/3$, as claimed, and $\mathbb{P}(D_1 \geq 3) = 1/3$. Iterating this argument, we get that

$$\mathbb{P}(D_1 \geq k) = \prod_{i=3}^{k} \mathbb{P}(k_i = i) = \prod_{i=3}^{k} \frac{1}{i} = \frac{2}{k!},$$

and thus

$$\mathbb{P}(D_1 = k) = \mathbb{P}(D_1 \geq k) - \mathbb{P}(D_1 \geq k+1) = \frac{2}{k!} - \frac{2}{(k+1)!} = \frac{2k}{(k+1)!},$$

as claimed.

By definition of our sampling algorithm, we know that the $(\ell+1)$-th internal node is inserted into the tree during the loop number $i = D_1 + \cdots + D_\ell + 1 = \bar{D}_\ell + 1$. Therefore, we get

$$\mathbb{P}(D_{\ell+1} = 2 | D_1, \ldots, D_\ell) = \mathbb{P}(k_{i+1} \neq i+1) = 1 - \frac{1}{\bar{D}_\ell + 2}, \text{ as claimed,}$$

and

$$\mathbb{P}(D_{\ell+1} \geq 3 | D_1, \ldots, D_\ell) = \frac{1}{\bar{D}_\ell + 2}.$$

Iterating this argument, we get that, for all $k \geq 3$,

$$\mathbb{P}(D_{\ell+1} \geq k | D_1, \ldots, D_\ell) = \prod_{j=\bar{D}_\ell+2}^{\bar{D}_\ell+k-1} \mathbb{P}(k_j = j) = \prod_{j=\bar{D}_\ell+2}^{\bar{D}_\ell+k-1} \frac{1}{j} = \frac{(\bar{D}_\ell+1)!}{(\bar{D}_\ell+k-1)!}.$$

This concludes the proof because

$$\begin{aligned}\mathbb{P}(D_{\ell+1} = k | D_1, \ldots, D_\ell) &= \mathbb{P}(D_{\ell+1} \geq k | D_1, \ldots, D_\ell) - \mathbb{P}(D_{\ell+1} \geq k+1 | D_1, \ldots, D_\ell) \\ &= \frac{(\bar{D}_\ell+1)!}{(\bar{D}_\ell+k-1)!} - \frac{(\bar{D}_\ell+1)!}{(\bar{D}_\ell+k)!} = \frac{(\bar{D}_\ell+1)!(\bar{D}_\ell+k-1)}{(k+\bar{D}_\ell)!},\end{aligned}$$

as claimed. □

Proof of Theorem 4.2.27. For this proof, we use the fact that the increasing Schröder tree is equal in distribution to the random $\boldsymbol{D}$-ary tree (see Lemma 4.2.29). For the lower bound, we use Lemma 4.2.26 and the fact that, almost surely for all $\ell \geq 1$, $H_\ell \geq Y_{\bar{D}_\ell+1}$, where we recall that Y_n is the depth of the leftmost leaf in an n-leaf uniform increasing Schröder tree and $\bar{D}_\ell = \sum_{i=1}^{\ell} D_i$. By Lemma 4.2.26, we have that, for all $\varepsilon > 0$,

$$\mathbb{P}(H_n \leq (1-\varepsilon)\ln n) \leq \mathbb{P}(Y_n \leq (1-\varepsilon)\ln n) \leq \mathbb{P}\left(\frac{Y_n - \ln n}{\sqrt{\ln n}} \leq -\varepsilon\sqrt{\ln n}\right) \to 0,$$

when n tends to infinity, which concludes the proof for the lower bound.

The proof for the upper bound is an adaptation of Devroye [Dev90] in

which the case of regular trees is treated (in regular trees, nodes have all the same degree they are also known as random k-ary trees). We denote by $N_1(n), \ldots, N_{D_1}(n)$ the sizes of the subtrees of the root of $\tau_n^{(\boldsymbol{D})}$; a straightforward adaptation of [Dev90, Lemma 2] gives that, conditionally on D_1,

$$\mathbb{P}((n-m+2)S_1 \geq x) \leq \mathbb{P}(N_1(n) \geq x) \leq \mathbb{P}(nS_1 \geq x), \tag{4.16}$$

where S_1 is the minimum of $D_1 - 1$ i.i.d. random variables uniform on $[0,1]$. We reason conditionally on the sequence $\boldsymbol{D}$ of random degrees, and denote by $\mathbb{P}_{\boldsymbol{D}}$ the law under this conditioning. We denote by $S_1, \ldots, S_{D_1}$ the spacings induced on $[0,1]$ by a sample of $D_1 - 1$ i.i.d. random variables uniform on $[0,1]$. Using the fact that the sizes of the subtrees of the root, $N_1(n), \dots, N_{D_1}(n)$ all have the same distribution, we get

$$\begin{aligned}\mathbb{P}_{\boldsymbol{D}}(H_n \geq k) &\leq \sum_{i=1}^{D_1} \mathbb{P}_{\boldsymbol{D}}(H_{N_i(n)} \geq k-1) = D_1 \mathbb{P}_{\boldsymbol{D}}(H_{N_1(n)} \geq k-1) \\ &\leq D_1 \mathbb{P}_{\boldsymbol{D}}(H_{nS_1} \geq k-1),\end{aligned}$$

where we have used Equation (4.16) in the last inequality. We now iterate this identity: we denote by $I(n) = n\prod_{i=1}^k S(D_i)$, where, for all $d \geq 2$, $S(d)$ is the minimum of $d-1$ i.i.d. random variables uniform on $[0,1]$. We get

$$\mathbb{P}_{\boldsymbol{D}}(H_n \geq k) \leq \left(\prod_{i=1}^k D_i\right) \mathbb{P}_{\boldsymbol{D}}\left(H_{I(n)} \geq 0\right) = \left(\prod_{i=1}^k D_i\right) \mathbb{P}_{\boldsymbol{D}}\left(n\prod_{i=1}^k S(D_i) \geq 1\right),$$

because a tree has height at least 1 as soon as it has at least one internal node. We now use Chebychev's inequality, which implies that, for all $\alpha \geq 1$,

$$\mathbb{P}_{\boldsymbol{D}}(H_n \geq k) \leq \left(\prod_{i=1}^k D_i\right) n^\alpha \mathbb{E}_{\boldsymbol{D}}\left[\prod_{i=1}^k S(D_i)^\alpha\right] = n^\alpha \prod_{i=1}^k \left(\frac{\Gamma(D_i+1)}{\prod_{i=1}^{D_i-1}(\alpha+i)}\right).$$

See [Dev90, Equation (1)] for the last equality. For all $\alpha \geq 1$, and for all $d \geq 2$, we have

$$\begin{aligned}\ln\Gamma(d+2) - \sum_{i=1}^{d} \ln(\alpha+i) &= \ln\Gamma(d+1) - \sum_{i=1}^{d-1}\ln(\alpha+i) + \ln(d+1) - \ln(\alpha+d) \\ &\leq \ln\Gamma(d+1) - \sum_{i=1}^{d-1}\ln(\alpha+i).\end{aligned}$$

Therefore, since $D_i \geq 2$ almost surely for all $i \geq 1$, we get

$$\mathbb{P}_{\boldsymbol{D}}(H_n \geq k) \leq n^\alpha \prod_{i=1}^k \left(\frac{\Gamma(3)}{\alpha+1}\right) = n^\alpha \left(\frac{2}{\alpha+1}\right)^k.$$

This expression is minimised for $\alpha = k/\ln n - 1$; taking $k = c\ln n$ and $\alpha = c-1$, we get that, for all $c > 0$,

$$\mathbb{P}_{\boldsymbol{D}}(H_n \geq c\ln n) \leq n^{c-1+c\ln(2/c)}.$$

If we take $c > \gamma$ where $\gamma = \inf\{c > 0 \colon c - 1 + c\ln(2/c) < 0\}$, then

$$\mathbb{P}_{\boldsymbol{D}}(H_n \geq c\ln n) \underset{n\to\infty}{\to} 0,$$

which concludes the proof for the upper bound. □

4.3 Strict monotonic Schröder trees

4.3.1 The model and its context

In this section we introduce and study a generalisation of the increasing Schröder trees, which we call *strict monotonic Schröder trees*. The main difference between the two models is that in strict monotonic Schröder trees, several internal nodes can be labelled by the same integer as long as they are not on the same ancestral line:

Definition 4.3.1. *A strict monotonic Schröder tree is a classical Schröder tree structure whose internal nodes are labelled by the integers between* 1 *and* ℓ *(for some* $\ell \geq 1$*), in such a way that each integer in* $\{1, \dots, \ell\}$ *appears at least once in the tree and the sequence of labels in the path from the root to any leaf is (strictly) increasing.*

In other words, it is a rooted labelled tree (only internal nodes are labelled) and the labelling is ***strict monotonic*** as in Definition 3.5.24. Remark that the trees are qualified by "strict" in the sense that the sequence of labels along the paths from the root to any leaf is strictly increasing.

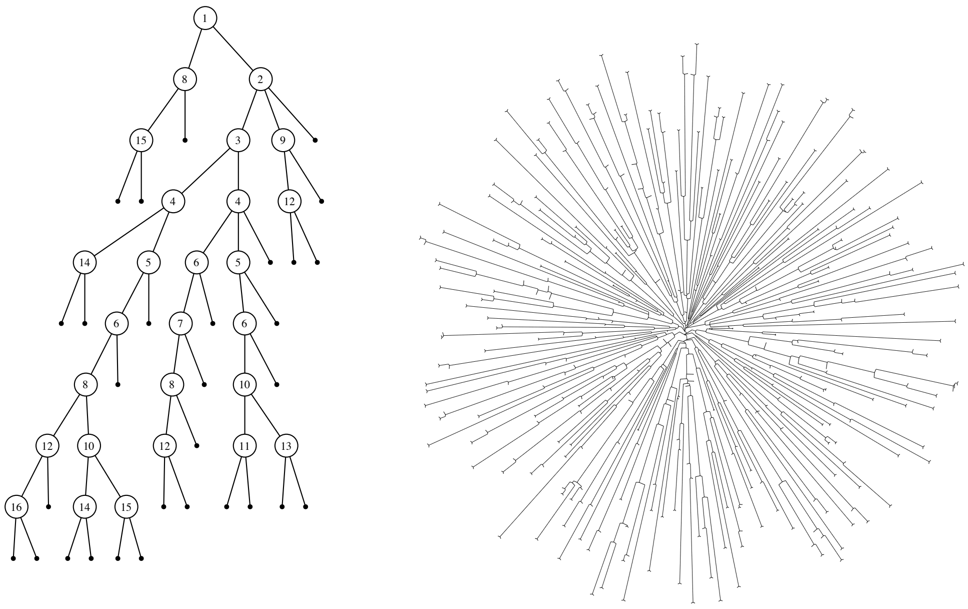


Figure 4.9: Two strict monotonic Schröder trees

In Figure 4.9 we show two strict monotonic trees: the left-hand-side one is of size 30 with 16 distinct labels, the right-hand-side one is of size 500 (sampled uniformly at random among all trees of size 500), with 495 internal nodes labelled with 372 distinct labels.

Because of the possible repetition of labels, this class of labelled trees cannot be directly specified using the classical analytic combinatorics operators for labelled structures. However, the following recursive construction allows us to specify the class of strict monotonic Schröder trees using operators for unlabelled structures. Every strict monotonic Schröder tree can be built as follows:

- *Start with a single (unlabelled) leaf.*

- *At step each step ℓ (for $\ell \geq 1$), select a non-empty subset of leaves and replace each of them by an internal node with label ℓ attached to a sequence of at least two leaves.*

4.3.2 Overview of the main results

As we have done before, we will start this section by solving the counting problem of this new family of trees in Section 4.3.3 and its asymptotics. We will then notice that the enumeration problem corresponds to the of *Ordered set partitions*. So that in Section 4.3.4 we will exhibit two bijections with *Ordered Set partitions*. Then in Section 4.3.4.3 we will show how *strict monotonic Schröder* trees can be generated from *increasing Schröder* by using *Eulerian numbers* and *runs* in permutations. We conclude the study by analysing some parameters on the trees where the results are summarised in Table 4.7.

	Mean	Variance	Limit law
Internal nodes	$n - \ln 2 \ln n$		
Distinct labels	$\frac{1}{2\ln 2}\, n$	$\frac{(1-\ln 2)}{(2\ln 2)^2}\, n$	Normal
Degree of the root	$2\ln 2 + 1$	$-2\,\ln 2\,(\ln 2 - 1)$	(shifted) zero-truncated Poisson
Depth of the leftmost leaf	$\ln n$	$\ln n$	Normal

Table 4.7: Summary of the main results on parameters of *Strict monotonic Schröder* trees

4.3.3 Enumeration and relationship with ordered Bell numbers

Using the iterative construction described above, we deduce the following specification for the class $\mathcal{G}$ of all strict monotonic Schröder trees:

$$\mathcal{G} = \mathcal{Z} \cup \big(\mathcal{G}[\mathcal{Z} \to (\mathcal{Z} \cup \mathrm{Seq}_{\geq 2}\,\mathcal{Z})]\big) \setminus \mathcal{G}.$$

Note that again the labelling is transparent and does not appear directly in the specification. The combinatorial meaning of this specification is the following: A tree of $\mathcal{G}$ is either a single leaf, or it is obtained by taking an already constructed tree in $\mathcal{G}$, and replace each leaf by either a leaf (i.e. no change) or an internal node attached to a sequence of at least two leaves. Furthermore we omit the case where no leaf is changing (this is why we subtract the set $\mathcal{G}$). Note that subtracting $\mathcal{G}$ is important, otherwise some integer values could be absent in the final tree. For example, if there is no change at step 2 but then the evolution continues, then 2 would not appear in the final tree but larger integers would appear as labels.

Using the symbolic method, we can translate this specification into a functional equation (with substitution) for the ordinary generating series:

$$G(z) = z + G\left(z + \frac{z^2}{1-z}\right) - G(z) = z + G\left(\frac{z}{1-z}\right) - G(z). \tag{4.17}$$

From this equation we extract the recurrence for the number g_n of strict monotonic Schröder trees with n leaves: we get

$$\begin{aligned} g_n = [z^n]G(z) &= [z^n]\left(z + G\left(z + \frac{z^2}{1-z}\right) - G(z)\right) \\ &= \delta_{n,1} + [z^n]\sum_{\ell \geq 1} g_\ell \left(\frac{z}{1-z}\right)^\ell - g_n \\ &= \delta_{n,1} - g_n + \sum_{\ell \geq 1} g_\ell [z^{n-\ell}] \left(\frac{1}{1-z}\right)^\ell . \end{aligned}$$

We use Kronecker's notation: $\delta_{n,1} = 1$ if $n = 1$ and 0 otherwise. The last coefficient extraction is similar to the integer composition (see [**FS09**, Example I.3, p. 44]). This implies

$$g_n = \begin{cases} 1 & \text{if } n = 1, \\ \sum\limits_{\ell=1}^{n-1} \binom{n-1}{\ell-1} g_\ell & \text{otherwise.} \end{cases} \tag{4.18}$$

The first coefficients are equal to a shift of the sequence of ordered Bell numbers (also called Fubini numbers or surjection numbers) referenced as EIS A000670:

$(g_n)_{n\in\mathbb{N}} = 0, 1, 1, 3, 13, 75, 541, 4683, 47293, 545835, 7087261, 102247563, 1622632573, \ldots$

The 13 trees of size 4 are depicted in Figure 4.10, we notice that a new tree with two repetitions of the label 2 appears. The rest of the trees are the same as the ones presented in Figure 4.3.

We recall that the n-th ordered Bell number counts the number of ordered partitions of a set of size n, where an ordered partition of a set $\mathcal{S}$ is an ordered sequence of disjoint subsets of $\mathcal{S}$ whose union is equal to $\mathcal{S}$. Ordered Bell numbers. This combinatorial class has been presented in Section 3.2. We recall here its specification

$$B = \text{Seq}\left(\text{Set}_{\geq 1} \mathcal{Z}\right). \tag{4.19}$$

Motivated by this remark, we define in Section Section 4.3.4.1 a bijection between the set of strict monotonic Schröder trees and the set of ordered partitions.

Following the approach developed by Pippenger in [**Pip10**] for ordered Bell numbers, we compute the exponential generating function of $\mathcal{G}$, i.e. we apply the Borel transform on $G(z)$. But first let us recall some basic properties of the latter transform. The *Borel transform*, which $\mathcal{B}$ denotes, takes as an argument an ordinary generating function and gives as its image the corresponding exponential generating series. More precisely, for all real-valued sequence $(a_n)_{n\geq 0}$, we set

$$\mathcal{B}\left[\sum_{n\geq 0} a_n z^n\right] = \sum_{n \geq 0} a_n \frac{z^n}{n!}.$$

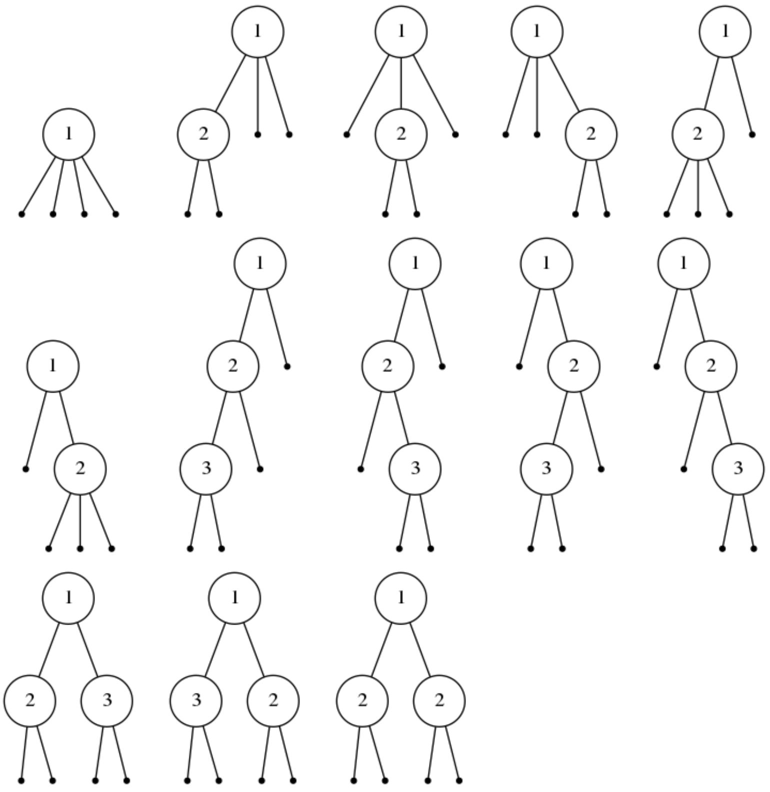


Figure 4.10: All 13 Strict monotonic Schröder trees of size 4

Note that if $t_n \leq \rho^n\, n!$ for n sufficiently large then $\mathcal{B}T(z)$ is analytic around 0. It is easy to check that:

Fact 4.3.2. *For all ordinary generating function $f = f(z)$, we have*

$$(i)\ \mathcal{B}\left[zf(z)\right] = \int_0^z \mathcal{B}(f)(t)\mathrm{d}t \qquad \textit{and} \qquad (ii)\ \mathcal{B}\left[f'(z)\right] = (\mathcal{B}[f(z)])' + z(\mathcal{B}[f(z)])''.$$

Proposition 4.3.3. *The exponential generating function enumerating strict monotonic Schröder trees is*

$$\mathcal{B}G(z) = \frac{1}{2}\left(z - \ln\left(2 - \mathrm{e}^z\right)\right).$$

Proof. Using Equation (4.18) and the fact that $g_0 = 1$, we obtain

$$g_n = \delta_{n,1} + \sum_{\ell=1}^{n-1} \binom{n-1}{\ell-1}\, g_\ell.$$

Adding g_n to both sides (multiplied by $\binom{n-1}{n-1} = 1$ on the right-hand side) gives

$$2g_n = \delta_{n,1} + \sum_{\ell=1}^{n} \binom{n-1}{\ell-1}\, g_\ell.$$

This recurrence can be directly used to derive an equation for the exponential generating function of $\mathcal{G}$:

$$2\mathcal{B}G(z) = z + \sum_{n\geq 1}\sum_{\ell=1}^{n} \binom{n-1}{\ell-1} g_\ell \frac{z^n}{n!},$$

which is the classical equation satisfied by the ordered Bell numbers. Following the approach of [Pip10], we differentiate the equation with respect to z and get

$$2(\mathcal{B}G(z))' = 1 + \sum_{n\geq 1}\sum_{\ell=1}^{n} \binom{n-1}{\ell-1} g_\ell \frac{z^{n-1}}{(n-1)!}.$$

Since the sum is the convolution of $\mathcal{B}G'(z)$ with $\exp(z)$, we get

$$(\mathcal{B}G(z))' = \frac{1}{2-\mathrm{e}^z},$$

which implies $\mathcal{B}G(z) = (z - \ln(2-\mathrm{e}^z))/2$ as claimed. □

Recall that ordered Bell numbers are specified by $\mathcal{B} = \mathrm{Seq}(\mathrm{Set}_{\geq 1}\,\mathcal{Z})$ and thus have exponential generating function $B(z) = 1/(2-\mathrm{e}^z)$. This directly implies that our sequence $(g_n)_{n\geq 0}$ is equal to the sequence of ordered Bell numbers shifted by one, since $B(z)$ is the derivative of $\mathcal{B}G(z)$. This link between strict monotonic trees and ordered Bell numbers has the interesting following consequence: we have shown that the (shifted) ordinary generating function of the ordered Bell numbers satisfies Equation (4.17). As far as we can tell, this was not known before.

The asymptotic behaviour of ordered Bell numbers is known (see, e.g., [FS09, p. 109]): if we denote by b_n the n-th ordered Bell number, then

$$b_n = \sum_{\ell=0}^{n} \ell! \left\{ {n \atop \ell} \right\} \underset{n\to\infty}{\sim} \frac{n!}{2\,(\ln 2)^{n+1}},$$

where the $\left\{ {n \atop \ell} \right\}$'s are the Stirling partition numbers (also called Stirling numbers of the second kind, see [FS09, Appendix A.8]). They count the number of ways to partition a set of n objects into k non-empty subsets.

Theorem 4.3.4. *The number b_n is equal to the number g_{n+1} of strict monotonic Schröder trees of size $n+1$, which implies that, for all $n \geq 1$,*

$$g_n = \sum_{\ell=0}^{n-1} \ell! \left\{ {n-1 \atop \ell} \right\} \underset{n\to\infty}{\sim} \frac{(n-1)!}{2\,(\ln 2)^{n}}.$$

4.3.4 Bijections with ordered Bell numbers and relationship to internal nodes

4.3.4.1 Bijection with Ordered Bell numbers based on the runs

Since the number of strict monotonic Schröder trees of size $n+1$ is equal to the number of ordered partitions of a set of size n, it is natural to try to find an explicit bijection between the two classes. In this section, we exhibit such a bijection.

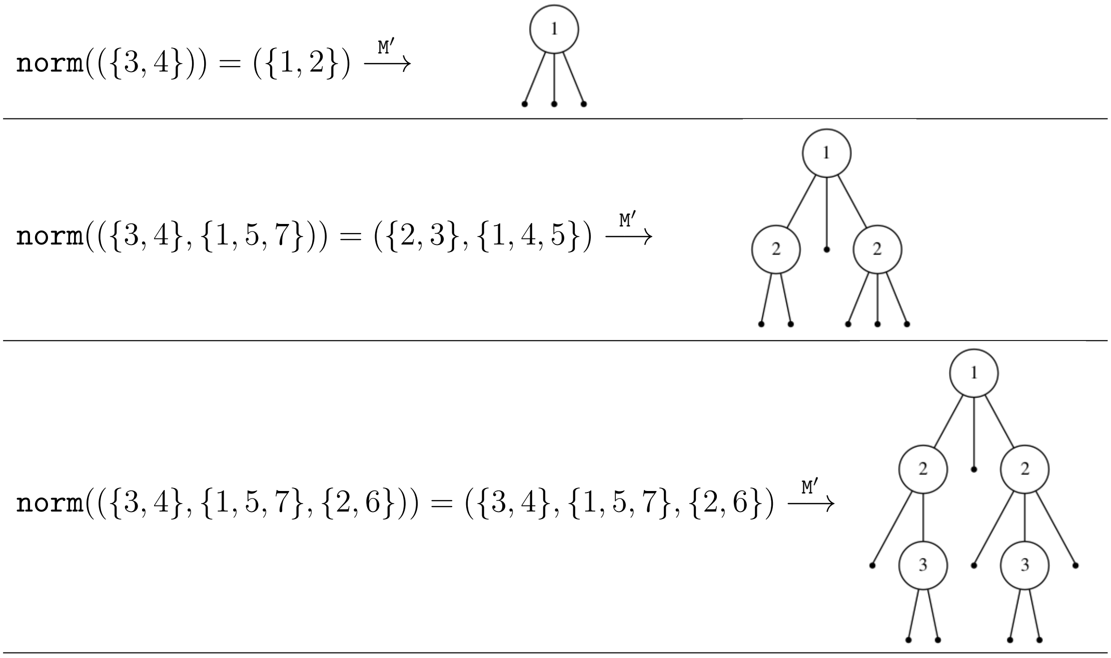
To describe precisely the bijection we need the following definitions and notations. Recall that the the subsets of an ordered partitions are ordered but the elements inside each subset are not. In the following, we denote by $p = (p_1, p_2, \ldots, p_\ell)$ the ordered partition of ordered subsets $p_1, \ldots, p_\ell$; for example, $(\{3,4\},\{1,5,7\},\{2,6\}) \neq (\{2,6\},\{3,4\},\{1,5,7\})$. We denote by $|p_i|$ the size of the i-th subset of p, and by $|p| = \sum_{i=1}^{\ell} p_i$ its total size (i.e. the number of elements of $\cup_{i=1}^{\ell} p_i$). Let $a = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ (with $r \geq 1$) be a subset of $\mathbb{N}$; without loss of generality, we can assume that $\alpha_1 < \alpha_2 < \ldots < \alpha_r$. A `run` of a is a maximal sequence $(\alpha_i, \alpha_{i+1}, \ldots, \alpha_j)$ $(1 \leq i \leq j \leq r)$ of consecutive integers, i.e. $(\alpha_i, \alpha_{i+1}, \ldots, \alpha_j) = (\alpha_i, \alpha_i + 1, \ldots, \alpha_i + j - i)$, $\alpha_{i-1} < \alpha_i - 1$ and $\alpha_{j+1} > \alpha_j + 1$. We define the function `runs` as the function that lists all the runs of a subset: for example, $\texttt{runs}(\{3,4\}) = (\{3,4\})$ and $\texttt{runs}(\{1,3,6,7\}) = (\{1\},\{3\},\{6,7\})$.

An ordered partition $p = (p_1, \ldots, p_\ell)$ is called *incomplete* if and only if $\cup_{i=1}^{\ell} p_i \neq \{1, 2, \ldots, |p|\}$, e.g. the partition $(\{3,4\},\{1,5,7\})$ is incomplete due to the fact that $\cup_{i=1}^{\ell} p_i = \{1,3,4,5,7\} \neq \{1,2,3,4,5\}$. We define the normalization of a partition p (either incomplete or not), denoted by $\texttt{norm}(p)$, as the ordered partition of $\{1, \ldots, |p|\}$ that keeps the relative order between the elements. For example, if $p = (\{3,4\},\{1,5,7\})$, then $\texttt{norm}(p) = [\{2,3\},\{1,4,5\}]$.

We are now ready to describe our bijection: we first define the mapping `M'`, which associates a strict monotonic Schröder tree to each (possibly incomplete) ordered partition $p = (p_1, \ldots, p_\ell)$. Before starting we fix an arbitrary order for the leaves in the tree once and for all (for example, the one given by the postorder traversal of the tree). Then The tree $\texttt{M'}(p)$ is the result of the following recursive procedure:

- At time zero, consider a tree with one internal node labelled by 1 to which are attached $|p_1| + 1$ leaves.
- At each time $2 \leq i \leq \ell$, we denote by $p'_1, \ldots, p'_i$ the ordered subsets of the renormalization of $(p_1, \ldots, p_i)$, i.e. $\texttt{norm}((p_1, \ldots, p_i)) = (p'_1, \ldots, p'_i)$. We denote by $r_1, \ldots, r_j$ the runs of p'_i, i.e. $\texttt{runs}(p'_i) = (r_1, \ldots, r_j)$; recall that each of $r_1, \ldots, r_j$ is a set of successive integers, possibly reduced to a singleton and iterate the following process: for k from 1 to j, take the leaf whose index is the first element of r_k and replace it with an internal node with label k attached to $|r_k| + 1$ leaves.

In Figure 4.11 we show how to construct $\texttt{M}'(p)$ when $p = (\{3,4\},\{1,5,7\},\{2,6\})$. The resulting strict monotonic Schröder tree is of size 8. It is straightforward to check that `M'` is indeed a bijection.



$$\mathtt{norm}((\{3,4\})) = (\{1,2\}) \xrightarrow{\mathtt{M}'}$$

$$\mathtt{norm}((\{3,4\},\{1,5,7\})) = (\{2,3\},\{1,4,5\}) \xrightarrow{\mathtt{M}'}$$

$$\mathtt{norm}((\{3,4\},\{1,5,7\},\{2,6\})) = (\{3,4\},\{1,5,7\},\{2,6\}) \xrightarrow{\mathtt{M}'}$$

Figure 4.11: The constructive bijection between an ordered partition and a strict monotonic Schröder tree

4.3.4.2 Bijection with Ordered Bell numbers based on the cycles

In Section 4.3.4.1 we exhibited a bijection between strict monotonic Schröder trees and ordered Bell numbers. This bijection thus implies that

$$\begin{aligned} G(\mathcal{Z}) &\cong \text{Seq}\,(\text{Set}_{\geq 1}\,\mathcal{Z}) \\ &\cong \text{Set}\,(\text{Cyc}\,(\text{Set}_{\geq 1}\,\mathcal{Z}))\,. \end{aligned}$$

To prove Proposition 4.3.12, we first exhibit the bijection $\mathtt{S}$ between the two latter sets. To do so, we first need to define better our notations: for example, since the order in a set is not relevant, i.e. $\{1,3,2\} = \{1,2,3\} = \{3,2,1\}$, we choose to always use the representation $\{i_1,\dots,i_m\}$ such that $i_1 < i_2 < \cdots < i_m$. Similarly, a cycle of sets, e.g. $(\{3,4\},\{1,5,6\},\{2\})$, is invariant by cyclic permutation of its elements, i.e.

$$C = (\{2,4\},\{1,5,6\},\{3\}) = (\{1,5,6\},\{3\},\{2,4\}) = (\{3\},\{2,4\},\{1,5,6\}).$$

In the following we choose to always use the representation such that the first element in the cycle contains 1: for our example, $C = (\{1,5,6\},\{3\},\{2,4\})$. Finally, given a set of cycles, we choose the representation in which the cycles are in decreasing lexicographic order: to each cycle, we associate the string of integers obtained from reading its elements from left to right, for example to $(\{1,5,6\},\{3\},\{2,4\})$, we associate 156324, and then order the cycles of sets according to this order. For example, the set $\{(\{1,2,4\},\{3\}),(\{1,3\},\{2,4,5\})\}$ has 2 cycles, the list of the string of the first one is 1243, the string of the second one is 13245. Since $3 > 2$, the canonical representation of this set of cycles is

$$\{(\{1,3\},\{2,4,5\}),(\{1,2,4\},\{3\})\}.$$

We are now ready to define the mapping $\mathtt{S}$: take S a set of cycles of sets of integers (in its canonical representation), and we denote by $X(S)$ the string of integers read from left to right in this canonical representation. E. g. if $S = \{(\{1,3\},\{2,4,5\}),(\{1,2,4\},\{3\})\}$, then $X(S) = 132451243$. Now define $\hat{X}(S)$ as a string of zeros of the same length as $X(S)$, and $c = 1$, and for all i between 1 and the maximum integer in S, go through the string $X(S)$ from right to left, i.e. for all j from length$(X(S))$ down to 1, if the digit in j-th position is a 1, replace the j-th digit in $\hat{X}(S)$ by c and increase c by 1. In our example, we eventually obtain $\hat{X}(S) = 264891375$. We denote by $s_1, \dots, s_m$ as the sizes of the sets of the cycles of S (in the order of the canonical representation); in our example, there are $m = 4$ sets in total (in the two cycles) and their sizes are $2, 3, 3, 1$. Define $\mathtt{S}(S)$ as the ordered partition having m parts of respective sizes $s_1, \dots, s_m$ and such that the elements of the first part are the first s_1 digits of $\hat{X}(S)$, the elements of the second part are the following s_2 digits of $\hat{X}(S)$ and so on. On our example, we get

$$\mathtt{S}(S) = (\{2,6\},\{4,8,9\},\{1,3,7\},\{5\}).$$

Lemma 4.3.5. *The mapping* $\mathtt{S}$ *is a one-to-one map from*

$$\mathrm{Set}(\mathrm{Cyc}(\mathrm{Set}_{\geq 1}\mathcal{Z})) \text{ onto } \mathrm{Seq}(\mathrm{Set}_{\geq 1}\mathcal{Z})$$

Proof. Given an ordered partition, i.e. a sequence os sets $S_1, \dots, S_m$. Denote by k the first integer in S_1 (since we use the canonical representation, it is also the smallest integer in S_1). And denote by i_j the integer such that S_{i_j} contains j, for all $1 \leq j \leq k-1$. Note that $1 = i_k < i_{k-1} < \dots < i_1$, and define $C_1 = (S_1, \dots, S_{i_{k-1}-1})$, $C_2 = (S_{i_{k-1}}, \dots, S_{i_{k-2}})$, until $C_k = (S_{i_1}, \dots, S_m)$. And set $\mathtt{S}^{-1}((S_1, \dots, S_m)) = \{C_1, \dots, C_k\}$. One can check that this is indeed the inverse of $\mathtt{S}$, which concludes the proof. □

Recall that, in Section 4.3.4.1, we have defined $\mathtt{M}'$, a bijection from the set of ordered partitions onto the set of strict monotonic Schröder trees. Therefore, $\mathtt{M}' \circ \mathtt{S}$ is a bijection from the $\mathrm{Set}(\mathrm{Cyc}(\mathrm{Set}_{\geq 1}\mathcal{Z}))$ onto the set of strict monotonic Schröder trees.

Lemma 4.3.6. *If* $X \in \mathrm{Set}(\mathrm{Cyc}(\mathrm{Set}_{\geq 1}\mathcal{Z}))$*, then the number of cycles of* X *is equal to the depth of the leftmost leaf of* $\mathtt{M}' \circ \mathtt{S}(X)$.

Proof. If X contains m cycles, then the integers $1, 2, \dots, m$ appear in reverse order and in different sets $s_m, s_{m-1}, \dots, s_1$ of the ordered partition $\mathtt{S}(X)$: s_i is the set containing the integer i for all $1 \leq i \leq m$. Moreover s_m is the first set in $\mathtt{S}(X)$ because the cycles are ordered in the canonical order. In the mapping $\mathtt{M}'$, s_m will form the root of the tree. Then s_{m-1} will create a node in the leftmost leaf, then s_{m-2} will create a node in the leftmost leaf and so on until s_1 is added to create a last node on the leftmost leaf. Thus the depth of the leftmost leaf is m. □

4.3.4.3 From increasing Schröder to strict monotonic Shröder

Another way of defining Ordered Bell numbers (or set partitions) defined in Section 3.2 is permutations with distinguished rises see [FS09, p.209]. A rise in a permutation is a pair of consecutive elements in permutation that is increasing. See Section 3.1.2 for a formal definition. A *cluster* is a maixmal sequence of adjacent rises. Let $\mathcal{C}$ be the class of clusters. By definition then a cluster has at least two elements. We have,

$$\mathcal{C} = \mathrm{Set}_{\geq 2}(\mathcal{Z}).$$

$$\boxed{3{,}4}, 6, 2, 5, 7, 8, \boxed{1}$$

Figure 4.12: A permutation with distinguished runs. By definition not all the rises need to be marked

In the labelled universe there is exactly one way of having n integers in an increasing order. Finally, if we let $\mathcal{PD}$ the class of permutations with distinguished runs. Here we don't require that all runs be distinguished. We can see an example in Figure 4.12.

Finally a permutation with distinguished rises is a sequence of elements that are either distinguished runs or simple integers and thus can be specified as follows:

$$\mathcal{PD} = \mathrm{Seq}(\mathcal{Z} + \mathcal{C}) = \mathrm{Seq}(\mathcal{Z} + \mathrm{Set}_{\geq 2}(\mathcal{Z})).$$

Which translates to:

$$PD(z) = \frac{1}{1 - (z + (e^z - 1 - z))} = \frac{1}{2 - e^z}.$$

Therefore $\mathcal{PD}$ is isomorphic to $\mathcal{B}$ the class of set partitions.

A very simple relation between *ordered Bell numbers* (*Ordered partitions*) and *runs* in *permutations* have been discovered by Velleman and Call in [**VC95**]. In our case this relation can be seen as a direct link between ***Increasing Schröder trees*** and ***Strict monotonic Schröder trees***. We recall that b_n is the n-th ordered Bell number:

$$b_n = \sum_{k=0}^{n-1} 2^k \left\langle {n \atop k} \right\rangle. \tag{4.20}$$

Where the numbers $\left\langle {n \atop k} \right\rangle$ are the *eulerian numbers* which have been introduced in Section 3.1.2. These numbers count the number of n-permutations having k descents and therefore they have $n - 1 - k$ rises.

Since *Eulerian numbers* are symmetric it is possible to rewrite Equation (4.20) as follows:

$$b_n = \sum_{k=0}^{n-1} 2^{n-k-1} \left\langle {n \atop k} \right\rangle. \tag{4.21}$$

Equation (4.21) has an easy combinatorial explanation. The idea is that *Eulerian numbers* count permutations where all the rises are distinguished. To obtain an *Ordered partition*, all we have to do is to allow some rises not to be distinguished. We take an n-permutation with k desents and puts dots between each rise. Therefore, since we have $n-1-k$ rises. We have

$[\{4\},\{6\},\{1\},\{2\},\{5\},\{3\}]$	$[\{4,6\},\{1\},\{2\},\{5\},\{3\}]$	$[\{4\},\{6\},\{1,2\},\{5\},\{3\}]$
$[\{4\},\{6\},\{1\},\{2,5\},\{3\}]$	$[\{4,6\},\{1,2\},\{5\},\{3\}]$	$[\{4,6\},\{1\},\{2,5\},\{3\}]$
$[\{4\},\{6\},\{1,2,5\},\{3\}]$	$[\{4,6\},\{1,2,5\},\{3\}]$	

Figure 4.13: The permutation $(4,6,1,2,5,3)$ has 3 rises, it gives 2^3 ordered partitions.

2^{n-k-1} possibilities of choosing configurations of these dots by taking them or not. Then, for each configuration we merge the element where the dots have been taken and put them in the same of the final *ordered partition*. For example, the permutation $(3,1,2)$ has one rise. We have two possibilities, putting the dot or not which give $(3,1\dot{2})$ and $(3,1,2)$. The first element gives $[\{3\},\{1,2\}]$ and the second gives $[\{3\},\{1\},\{2\}]$. Another example is depicted in Figure 4.13.

If we look back at the bijection on *Increasing Schröder* trees presented in Section 4.2.5.1 with the mapping $\mathcal{M}$. It is possible to do a mirror mapping $\hat{\mathcal{M}}$, for permutations where 2 comes before 1. For instance if $n = \sigma_1$ then we add a node to the last internal node of the tree. In the other case, we enumerate the leaves from left to right and add a new binary node. As a consequence of this mapping. Let $\mathcal{MHP}$ be the class of permutations where 2 appears before 1. then,

Lemma 4.3.7. *Let σ be an n-permutation and $\sigma \in \mathcal{HP}$ (i.e 1 appears before 2). Then $\hat{\mathcal{M}}(mir(\sigma))$ gives the same tree.*

Proof. Each permutation $\sigma \in \mathcal{HP}$ has its mirror permutation $mir(\sigma) \in \mathcal{MHP}$. The proof follows directly, since the mapping $\hat{\mathcal{M}}$ is the mirror mapping of $\mathcal{M}$. □

Finally, if we have an *Increasing Schröder* tree t which its corresponding permutation σ has k descents. Then, the tree corresponding to $\hat{\mathcal{M}}(\sigma)$ has $n-k-1$ descents. This allows us to have a tree representation for all permutations (without restrictions). Each *Increasing Schröder* tree being the image of two separate permutations.

Corollary 4.3.8. *As a conclusion, an Increasing Schröder tree t of size n which its corresponding permutation $\sigma = \mathcal{M}^{-1}(t)$ has k descents, makes*

$$2^{n-k-1} + 2^k$$

strict monotonic Schröder trees of size $n-1$ (remember that Ordered Bell numbers are size shifted with strict monotonic Schröder trees).

Open question *(Tree interpretation). It is an open question to give in interpretation in the tree world on the kind of Strict monotonic Schröder trees that are created with an Increasing Schröder tree.*

4.3.5 Analysis of typical parameters

In this section, we give information about the shape of a *typical* strict monotonic Schröder tree: more precisely, we prove limit theorems for the number of distinct labels, the number of internal nodes and the arity of the root in a tree picked uniformly at random among all strict monotonic Schröder trees of size n (i.e. with n leaves).

4.3.5.1 Quantitative analysis of the number of iteration steps

The main novelty of strict monotonic Schröder trees compared to increasing Schröder trees is that repetitions of labels are allowed: it is thus natural to ask how many repetitions there are in a *typical* strict monotonic Schröder tree. To answer this question, on can mark iterations by adding a new variable u in Equation (4.17):

$$G(z,u) = z + u\, G\left(\frac{z}{1-z}, u\right) - u\, G(z,u),$$

which implies

$$g_{n,k} = \begin{cases} 1 & \text{if } n = 1 \text{ and } k = 0, \\ \sum\limits_{\ell=1}^{n-1} \binom{n-1}{\ell-1}\, g_{\ell,k-1} & \text{otherwise,} \end{cases} \tag{4.22}$$

with n being the size and k the number of iteration steps (i.e. the number of distinct labels). In Figure 4.14, we show the first values of $(g_{n,k})$ that are stored in EIS A019538.

1,						
0,	1,					
0,	1,	2,				
0,	1,	6,	6,			
0,	1,	14,	36,	24,		
0,	1,	30,	150,	240,	120,	
0,	1,	62,	540,	1560,	1800,	720

Figure 4.14: Distribution of $(g_{n,k})_k$ for $n \in \{1,\ldots,7\}$

This recurrence is analogous to the one relating ordered Bell numbers and Stirling partition numbers (see Equation (4.18)).

Theorem 4.3.9. *The number of strict monotonic Schröder trees of size n with exactly k distinct labels is given by*

$$g_{n,k} = k! \left\{ {n+1 \atop k} \right\}.$$

We denote by $X_n^{\mathcal{G}}$ the number of distinct labels in a tree picked uniformly at random among all strict monotonic Schröder trees of size n: for all $n \geq 1$, $X_n^{\mathcal{G}}$ is a random variable such that $\mathbb{P}(X_n^{\mathcal{G}} = k) = g_{n,k} / \sum_{k=1}^{n} g_{n,k}$. Then, asymptotically when n tends to infinity,

$$\frac{X_n^{\mathcal{G}} - \frac{n}{2\ln 2}}{\sqrt{\frac{(1-\ln 2)n}{(2\ln 2)^2}}} \xrightarrow{d} \mathcal{N}(0,1).$$

The analysis of the limiting distribution is classical in the quasi-powers framework established by Hwang [**Hwa98**]; see [**FS09**, p. 645, 653] for details and applications.

Proof. Recall that $g_{n,k} = k!\left\{ {n+1 \atop k} \right\}$ is the number of ordered partitions of a set of size n having k non-empty parts. It is known (see, e.g. [**Ben73**, Example 3.4]) that, if K_n is the number of parts in an ordered set partition of size n, then

$$\frac{K_n - \frac{n}{2\ln 2}}{\sqrt{\frac{(1-\ln 2)n}{(2\ln 2)^2}}} \xrightarrow{d} \mathcal{N}(0,1),$$

in distribution. This concludes the proof since K_n has the same distribution as $X_n^{\mathcal{G}}$ for all $n \geq 1$. □

4.3.5.2 Quantitative analysis of the number of internal nodes

In this model the number of internal nodes is different from the number of distinct labels that appear in the tree: this is because one integer can label several internal nodes. It is thus natural to ask how many internal nodes a typical strict monotonic Schröder trees of size n (i.e. with n leaves) has. The specification marking both leaves (with variable z) and internal nodes (with variable u) is

$$G(z,u) = z + G\left(z + \frac{uz^2}{1-z}, u\right) - G(z,u). \tag{4.23}$$

We recall that the substitution $z \to z + \frac{uz^2}{1-z}$ means that at each iteration each leaf can be left as it is ($z \to z$) or expanded into an internal node attached to an arbitrary number of leaves ($z \to \frac{z^2}{1-z}$). A new internal, marked with the variable u, is created only in the second case.

1,						
0,	1,					
0,	1,	2,				
0,	1,	5,	7,			
0,	1,	9,	31,	34,		
0,	1,	14,	86,	226,	214	
0,	1,	20,	190,	874,	1946,	1652

Figure 4.15: Distribution of $(g_{n,k})_k$ for $n \in \{1,\dots,7\}$

For all $1 \leq n$ and $1 \leq k \leq n-1$, we denote by $g_{n,k}$ the number strict monotonic Schröder trees with n leaves and k internal nodes: Figure 4.15 shows the values of $(g_{n,k})_{1\leq k\leq n-1}$ for $n \in \{1,2,\dots,7\}$. This triangle of integers is not yet stored in OEIS. However, its diagonal is equal to EIS A171792. In fact in the diagonal the numbers corresponds to the number of strict monotonic trees with n leaves and $n-1$ internal nodes, i.e. binary strict monotonic trees: this class of trees is studied in [**BGGW20**].

Theorem 4.3.10. *If we denote by $I_n^{\mathcal{G}}$ the (random) number of internal nodes in a tree picked uniformly at random among all strict monotonic Schröder trees of size n, then, asymptotically when n tends to infinity,*

$$\mathbb{E}[I_n^{\mathcal{G}}] \underset{n\to\infty}{=} n - (\ln 2)(\ln n) + \frac{\pi^2}{12} - 1 + (\ln 2)\left(-\gamma + \frac{\ln 2}{2} + \ln\ln 2\right) + o(1),$$

where γ is the Euler-Mascheroni constant.

Proof. For all $n \geq 1$, we denote by $h_n = \sum_{k=1}^{n-1} k g_{n,k}$, and let H be the ordinary generating function of $(h_n)_{n\geq 1}$; we have

$$H(z) = \left(\frac{\partial G(z,u)}{\partial u}\right)_{|u=1}.$$

The ratio h_n/g_n is equal to the expected number of internal nodes in a tree taken uniformly at random among all strict monotonic Schröder trees of size n; we are thus interested in the asymptotic behaviour of this ratio. Differentiating according to u and then substituting u by 1 in Equation (4.23) gives

$$H(z) = \frac{z^2}{1-z} G'\left(\frac{z}{1-z}\right) + H\left(\frac{z}{1-z}\right) - H(z), \tag{4.24}$$

because

$$\left(\frac{\partial G(z,u)}{\partial z}\right)_{|u=1} = G'(z).$$

Since Equation (4.24) is similar to Equation (4.17), we apply the same method as in the proof of Proposition 4.3.3. We first derive

$$(\mathcal{B}H(z))' = \frac{1}{2-e^z}\left(\mathcal{B}\left[\frac{z^2}{1-z}G'\left(\frac{z}{1-z}\right)\right]\right)'.$$

Then using Equation (4.17) we deduce

$$\left(\mathcal{B}\left[\frac{z^2}{1-z}G'\left(\frac{z}{1-z}\right)\right]\right)' = -z + \frac{z^2}{2} + 2\left(\mathcal{B}\left[z^2(1-z)G'(z)\right]\right)'.$$

Furthermore since for any function F we have $\mathcal{B}zF(z) = \int_0^z \mathcal{B}F(t)\mathrm{d}t$, we can simplify the equation into

$$(\mathcal{B}H(z))' = \frac{1}{2-\mathrm{e}^z}\left(-z + \frac{z^2}{2} + 2\int_0^z \mathcal{B}G'(t)\,\mathrm{d}t - 2\int_0^z\int_0^t \mathcal{B}G'(u)\,\mathrm{d}u\,\mathrm{d}t\right).$$

Then, since $\int_0^z \mathcal{B}G'(t)\mathrm{d}t = z(\mathcal{B}G(z))'$, we obtain

$$\begin{aligned}(\mathcal{B}H(z))' &= \frac{1}{2-\mathrm{e}^z}\left(-z + \frac{z^2}{2} + 2z(\mathcal{B}G(z))' - 2\int_0^z t(\mathcal{B}G(t))'\mathrm{d}t\right)\\ &= \frac{1}{2-\mathrm{e}^z}\left(-z + \frac{z^2}{2} + \frac{2z}{2-\mathrm{e}^z} - 2\int_0^z \frac{t}{2-\mathrm{e}^t}\,\mathrm{d}t\right)\\ &= \frac{1/2}{1-\mathrm{e}^z/2}\left(-\frac{\pi^2}{12} + \frac{(\ln 2)^2}{2} - z\left(1 - \ln(1-\mathrm{e}^z/2) - \frac{1}{1-\mathrm{e}^z/2}\right) + \mathrm{Li}_2(\mathrm{e}^z/2)\right),\end{aligned}$$

where Li_2 is the dilogarithm function, defined in [FS09, section VI.8.]. Using its asymptotic development at 1, we get

$$(\mathcal{B}H(z))' \underset{z\to\ln 2}{\sim} \frac{1}{2\ln 2}\frac{1}{(1-z/\ln 2)^2}$$

$$- \left(\frac{1}{2 \ln 2} - \frac{\pi^2}{24 \ln 2} + \frac{\ln 2}{4} - \frac{\ln 2 + \ln \ln 2 + \ln (1 - z/ \ln 2)}{2} \right) \frac{1}{1 - z/ \ln 2}$$
$$- \frac{1}{2} - \frac{7 \ln 2}{24} + \frac{\pi^2}{48} + \frac{(\ln 2)^2}{8} + \frac{\ln 2 \ln \ln 2}{4} + O \left(\ln \left(\frac{1}{1 - z/ \ln 2} \right) \right).$$

By using classical transfer theorems we obtain the result by extracting the $(n-1)$-th coefficient of $(\mathcal{B}H(z))'$ and dividing it by the n-th coefficient of $\mathcal{B}G(z)$. □

4.3.5.3 Quantitative characteristics of the root node

In this section, we look at the arity of the root in a typical strict monotonic Schröder tree. We denote by $A_n^{\mathcal{G}}$ the arity of the root in a tree picked uniformly at random among all strict monotonic Schröder trees of size n, and by p_n its probability generating function:

$$p_n(u) = \sum_{k \geq 0} \mathbb{P}(A_n^{\mathcal{G}} = k) \, u^k.$$

Theorem 4.3.11. *Asymptotically when n tends to infinity, $A_n^{\mathcal{G}}$ converges in distribution to a (shifted) zero-truncated Poisson law with parameter* $\ln 2$*, i.e. for all $u \geq 0$,*

$$p_n(u) \underset{n \to \infty}{\to} u e^{u \ln 2} - u.$$

This implies that $\mathbb{E}[A_n^{\mathcal{G}}] \to 2 \ln 2 + 1$ *and* $\mathbb{V}[A_n^{\mathcal{G}}] \to -2 \, \ln 2 \, (\ln 2 - 1)$ *when n tends to infinity*

Proof. Thanks to the bijection of Section 4.3.4.1, we know that $A_n^{\mathcal{G}}$ is equal to the size of the first subset in an ordered partition picked uniformly at random among all ordered partitions of $\{1, \ldots, n-1\}$. We denote by $\mathcal{P}$ the class of ordered partitions, 1 is the empty partition, $\mathcal{Z}$ is a singleton, and $\mathcal{U}$ marks the elements in the first subset. Here the specification is defined in the context of labelled object, thus the associated generating functions are exponential (see [FS09] for notation details):

$$\mathcal{P} = 1 + \underset{\geq 1}{\text{Set}}(\mathcal{U}\mathcal{Z}) \star \text{Seq}(\underset{\geq 1}{\text{Set}}\, \mathcal{Z}).$$

Using the symbolic method for exponential generating function, we get

$$P(z,u) = 1 + \frac{e^{uz} - 1}{2 - e^z}.$$

Thus, if we set

$$\tilde{p}_n(u) = \frac{[z^n]P(z,u)}{[z^n]P(z,1)},$$

for all $n \geq 0$, then

$$[z^n]P(z,u) \underset{n \to \infty}{\to} \frac{1}{2} (2^u - 1) (\ln 2)^{-n-1}.$$

This implies that, for all $u \geq 0$,

$$\tilde{p}_n(u) \underset{n \to \infty}{\to} 2^u - 1.$$

Note that, by definition, $\tilde{p}_n(u)$ is the probability generating function of the size S_n of the first subset in an ordered partition picked uniformly at random among all ordered partitions of $\{1, \ldots, n-1\}$. Because of the bijection of Section 4.3.4.1, we know that $A_n^{\mathcal{G}}$ and S_{n-1} have the same distribution, implying that $p_n(u) = u\tilde{p}_n(u)$. This concludes the proof. □

4.3.5.4 Typical depth of the leftmost leaf

In this section, we prove a central limit theorem for the depth of the leftmost leaf in a typical strict monotonic Schröder tree:

Proposition 4.3.12. *Let $Y_n^{\mathcal{G}}$ be the depth of the leftmost leaf in a tree taken uniformly at random among all increasing Schröder trees of size n. In distribution when n tends to infinity, we have*

$$\frac{Y_n^{\mathcal{G}} - \ln n}{\sqrt{\ln n}} \xrightarrow{d} \mathcal{N}(0,1).$$

The depth of the leftmost leaf is a lower bound for the height (since the height of a tree is the maximal depth of its leaves), and it has the advantage of being easier to specify and analyse than the height itself.

Recall that, in Section 4.2.7.4, we proved a similar central limit theorem for the depth of the leftmost leaf in a typical increasing Schröder tree.

In this section, the variable $\mathcal{U}$ marks the depth of the leftmost leaf. Using the evolution process, we get that

$$G(z,u) = z + \left(\frac{G(y,u)}{y}\right)_{|y=\frac{z}{1-z}} \cdot \left(z + \frac{uz^2}{1-z}\right) - G(z,u).$$

At each iteration step we start by chopping off the leftmost leaf (this corresponds to $G(y,u)/y$). Each of the other leaves either stays unchanged or is replaced by an internal node to which is attached a sequence of at least two leaves (this corresponds to substituting y by $z/(1-z)$). Finally we put back the leaf that has been chopped off and there we have 2 choices, either it stays unchanged (z) or it is replaced by an internal node with at least two leaves attached to it ($z^2/(1-z)$) in which case we multiply by u because the depth of the leftmost leaf has been increased by one.

Iterating this specification, we can calculate the first coefficients (see Table 4.8): they are equal to the first coefficients of a shifted version of EIS A129062. From the specification it is possible to derive a recurrence relation on the coefficients $g_{n,k}$. We have $g_{1,0} = 1$ and for all $n \geq 2$ and $1 \leq k \leq n-1$,

$$g_{n,k} = \sum_{\ell=k+1}^{n-1} g_{\ell,k} \binom{n-2}{\ell-2} + \sum_{\ell=k}^{n-1} g_{\ell,k-1} \binom{n-2}{\ell-1}. \tag{4.25}$$

This equation can be interpreted combinatorially using the evolution process (this reasoning is similar to the one leading to Equation (4.18).): At each iteration step, the depth of the leftmost leaf either stays unchanged or increases by 1. Therefore each tree in $\mathcal{G}_{n,k}$, the set of all strict monotonic Schröder trees whose leftmost leaf is at depth k, was, before the last iteration, either a tree of $\mathcal{G}_{\ell,k}$ of a tree of $\mathcal{G}_{\ell,k-1}$, for some $\ell < n$. There are $\binom{n-2}{\ell-1}$ to expand a tree of $\mathcal{G}_{\ell,k-1}$ into a tree of $\mathcal{G}_{n,k}$ in one iteration step: it is the number of ways to partition the n leaves of the size-n tree into ℓ parts of size at least one (when a part is of size 1, the corresponding leaf in the ℓ-size node stays unchanged, otherwise, it becomes an internal node of out-degree the size of the part) in a way that the first part is of size at least 2 (the left-most leaf becomes an internal node attached to two leaves). Similarly, there are $\binom{n-2}{\ell-2}$ number of ways to expand a tree of $\mathcal{G}_{\ell,k}$ into a tree of $\mathcal{G}_{n,k}$ in one iteration step: it is the number of ways partition the n leaves of the size-n tree into ℓ parts of size at least one such that the first part is of size 1.

Using the bijection exhibited in Section 4.3.4.2 we have the following

Proposition 4.3.13. *The exponential generating function of $g_{n,k}$ is*

$$G(z,u) = \sum_{n\geq 0}\sum_{k\geq 0} g_{n,k} u^k \frac{z^n}{n!} = \int\limits_0^z \left(\frac{1}{2-\mathrm{e}^x}\right)^u \mathrm{d}x.$$

Proof. From the bijection, since the depth of the leftmost leaf is the number of cycles we get a direct specification by marking the cycles in the following

$$\mathcal{D} = \mathrm{Set}\left(\mathcal{U}\,\mathrm{Cyc}\left(\mathrm{Set}_{\geq 1}\,\mathcal{Z}\right)\right).$$

Therefore $D(z,u) = \exp\left(u \ln\left(\frac{1}{1-(\exp(z-1))}\right)\right)$. The number of trees of size n is the number of ordered Bell numbers of size $n-1$, so we integrate the last expression. □

The discussion above also leads to the following identity

Proposition 4.3.14. *For all $n \geq 2$ and $1 \leq k \leq n-1$,*

$$g_{n,k} = \sum_{m=0}^{n-1} \left\{ {n-1 \atop m} \right\} \left[{m \atop k} \right].$$

Where $\left[{n \atop k} \right]$ are Stirling Cycle numbers (also known as Stirling numbers of the first kind). They count the number of cycles of size k in a permutation of size n.

Proof. The proof is a direct consequence of the previous construction. The number of trees of size n with leftmost leaf at depth k can be constructed by looking at set partitions of size $n-1$ elements into i subsets for all possible sizes of i which is counted by $\left\{ {n-1 \atop i} \right\}$ then for each partition of size i we see how many cycles of size k we can build with it. □

Proof of Proposition 4.3.12. We can make the calculations on the bivariate generating function $D(z,u)$ which enters the scope of quasi-powers framework. Theorem IX.11 in [**FS09**, p. 669] is applicable. The exponent $\alpha(u) = u$ is analytic and $\alpha(1) = 1$ and it satisfies $\alpha'(1) + \alpha''(1) = 1 \neq 0$. So $D(z,u)$ is asymptotically Gaussian with mean and variance as announced. Finally the shift that we have between the size of trees and the ordered partitions does not affect the first orders since $\ln(n+1) \sim \ln n$. □

Open question (Height of trees). Unfortunately, we were not able to analyse the height of this tree model even though we strongly believe that the mean height is $c \ln n$ for some positive constant c.

1						
0	1					
0	2	1				
0	6	6	1			
0	26	36	12	1		
0	150	250	120	20	1	
0	1082	2040	1230	300	30	1

Table 4.8: Values of $g_{n,k}$, the number of n strict monotonic Schröder trees of size n with leftmost leaf at depth k, for all $0 \leq k \leq n \in \{1, \dots, 7\}$.

4.4 Strict monotonic general Schröder trees

In this section, we introduce a generalisation of the strict monotonic Schröder tree model of Section 4.3: the difference is that we allow internal nodes to have only one child (we call these nodes "unary" nodes). Since the size of a tree is the number of its leaves, allowing unary nodes without adding any other constraint would mean that there would be infinitely many trees of any given size n. To avoid this, we add the following constraint: at each growth step, at least one leaf is expanded as an internal node of arity greater or equal to 2.

4.4.1 The model and its enumeration

Definition 4.4.1. *A* strict monotonic general tree *is a labelled tree that can be obtained by the following evolution process:*

- *Start with a single (unlabelled) leaf.*
- *At every step $\ell \geq 1$, select a non-empty subset of leaves, replace all of them by internal nodes labelled by ℓ, attach to at least one of them a sequence of at least two leaves, and attach to all others a unique leaf.*

The two trees in Figure 4.16 are sampled uniformly among all strict monotonic general trees of respective sizes (i.e. number of leaves) 15 and 500. The left-hand-side tree has 14 distinct node-labels, i.e. it can be built in 14 steps using Definition 4.4.1. The right-hand-side tree is represented as a circular tree with stretched edges like in the right-hand-side of Figure 4.16. Here the tree contains 500 leaves built with 499 iterations of the growth process. But in comparison with the increasing and strict monotonic Schröder trees drawn in the latter sections and containing respectively 492 and 495 internal nodes, this one contains 62494 internal nodes, most of them being unary nodes.

4.4.2 Overview of the main results

Once again we start with the counting problem of this family of trees. We find a recurrence on the trees, but this time we have no link with classical objects in combinatorics. In order to find an asymptotic enumeration we look for an approximating model of this family of trees in

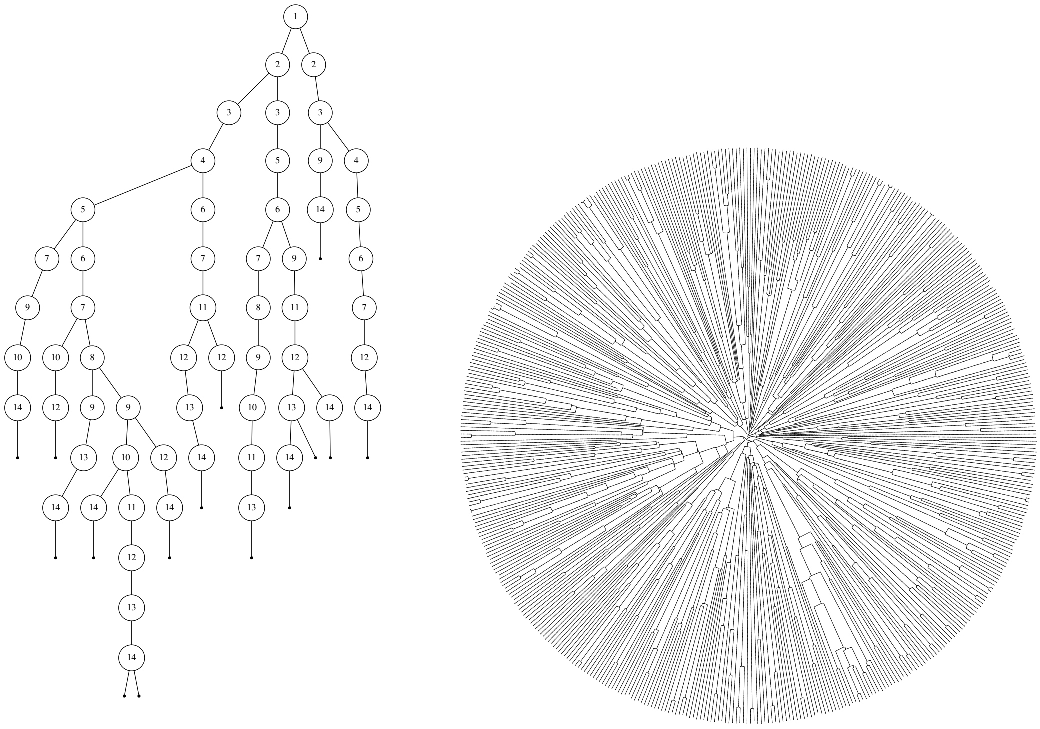


Figure 4.16: Two strict monotonic general trees

Section 4.4.3 by taking a subset of trees. Surprisingly, this approximating model will turn out to to be simple enough to have a close formula for enumeration and its asymptotic behaviour corresponds to the one of the general model (up to a constant). The approximating has also a very simple correspondence with *labelled graphs* that we exhibit in Section 4.4.4. Then, we will turn to study some parameters of these trees in Section 4.4.5. Unfortunately, we were not able to find limiting laws for this new model due to the explosion of the enumeration problem. It was not possible for us to get closed form of generating functions. However, we do find the order of growth of the main interesting parameters that we summarised in Table 4.1.

We can specify strict monotonic general trees using the symbolic method; once again the labelling is transparent and does not appear in the specification (i.e. we use ordinary generating functions). In this section, we denote by $F(z)$ the generating function of strict monotonic general trees and by $\mathcal{F}_n$ the set of all strict monotonic general trees of size n; from Definition 4.4.1, we get

$$F(z) = z + F\left(z + \frac{z}{1-z}\right) - F(2z). \tag{4.26}$$

The combinatorial meaning of this specification is the following: A tree of is either a single leaf, or it is obtained by taking an already constructed tree, and replace each leaf by either a leaf (i.e. no change) or an internal node attached to a sequence of at least one leaf. Furthermore we omit the case where no leaf is replaced by an internal node with at least to children (this is encoded in the subtracting $F(2z)$).

From this equation we extract the recurrence for the number f_n of strict monotonic general trees with n leaves. In fact we get

$$\begin{aligned} f_n = [z^n]F(z) = [z^n]\left(z + F\left(z + \frac{z}{1-z}\right) - F(2z)\right) \\ = \delta_{n,1} - 2^n f_n + [z^n] \sum_{\ell \geq 1} f_\ell \left(z + \frac{z}{1-z}\right)^\ell \\ = \delta_{n,1} - 2^n f_n + \sum_{\ell \geq 1} f_\ell \, [z^{n-\ell}] \sum_{i=0}^{\ell} \binom{\ell}{i} \left(\frac{1}{1-z}\right)^i, \end{aligned}$$

which implies that

$$f_n = \begin{cases} 1 & \text{if } n = 1, \\ \sum_{\ell=1}^{n-1} \sum_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i} \, 2^{\ell-i} \, \binom{n-\ell-1}{i-1} \, f_\ell & \text{for all } n \geq 2. \end{cases} \qquad (4.27)$$

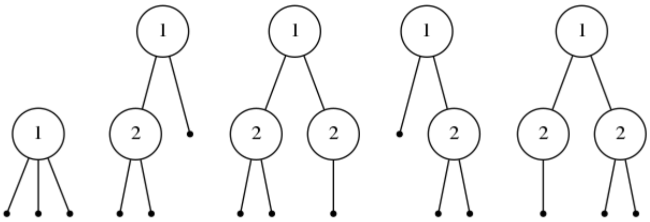


Figure 4.17: All strict monotonic generam Schröder trees of size 3

The combinatorial meaning of the inner sum is the following: starting with a tree of size ℓ we reach a tree of size n in one iteration by adding $n - \ell$ leaves. The index i in the inner sum stands for the number of leaves that are replaced by internal nodes or arity at least 2, by definition of the model (see Definition 4.4.1), we have $1 \leq i \leq \min(n - \ell, \ell)$. There are $\binom{\ell}{i}$ possible choices for the i leaves that are replaced by nodes of arity at least 2. Each of the remaining $\ell - i$ leaves is either kept unchanged or replaced by a unary node, which gives $2^{\ell-i}$ possible choices. And finally, there are $\binom{n-\ell-1}{\ell-1}$ possible ways to distribute the (indistinguishable) $n - \ell$ additional leaves among the i new internal nodes so that each of the i nodes is given at least one additional leaf (it already has one leaf, which is the leaf that was replaced by an internal node). The first terms of the sequence are the following:

$$(f_n)_{n \geq 0} = (0, 1, 1, 5, 66, 2209, 180549, 35024830, 15769748262, 16187601252857, \dots).$$

The 5 trees of size 3 are depicted in Figure 4.17.

Theorem 4.4.2. *There exists a constant* c *such that the number* f_n *of strict monotonic general trees of size* n *satisfies, asymptotically when* n *tends to infinity,*

$$f_n \underset{n\to\infty}{\sim} c\,(n-1)!\, 2^{\frac{(n-1)(n-2)}{2}}.$$

In the proof of the latter theorem we exhibit the following bounds $1.4991 < c < 1.8932$. But through several experimentations we see that $c < 3/2$ but it is close to it. For instance when $n = 1000$ we get $c \approx 1.49913911$. We postpone the proof to the next section to make use of the number of iteration steps.

4.4.3 Iteration steps and asymptotic enumeration of the trees

In this section, we look at the number of distinct internal-node labels that occur in a typical strict monotonic general tree, i.e. the number of iterations needed to build it:

Proposition 4.4.3. *Let $f_{n,k}$ denotes the number of strict monotonic general trees of size n with k distinct node-labels, then, for all $n \geq 1$,*

$$f_{n,n-1} = (n-1)!\, 2^{\frac{(n-1)(n-2)}{2}}.$$

Note that the first terms are

$$(f_{n,n-1})_{n\geq 0} = (0, 1, 1, 4, 48, 1536, 122880, 23592960, 10569646080, 10823317585920, \dots).$$

Proof. We use a new variable u to mark the number of iterations (i.e. the number of distinct node-labels) in the iterative Equation (4.27). We get

$$F(z,u) = z + u\, F\left(z + \frac{z}{1-z}, u\right) - u\, F(2z,u). \tag{4.28}$$

Using either Equation (4.28) or a direct combinatorial argument, we get that, for all $k \geq n$, $f_{n,k} = 0$ and

$$f_{n,k} = \begin{cases} 1 & \text{if } n = 1 \text{ and } k = 0, \\ \sum\limits_{\ell=k}^{n-1} \sum\limits_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i}\, 2^{\ell-i} \binom{n-\ell-1}{i-1} f_{\ell,k-1} & \text{if } 1 \leq k < n. \end{cases}$$

In particular, for $k = n-1$, we get

$$\begin{aligned} f_{n,n-1} &= (n-1)\, 2^{n-2}\, f_{n-1,n-2} = f_{1,0} \prod_{j=1}^{n-1} j 2^{j-1} = (n-1)! 2^{\sum_{j=0}^{n-2} j} \\ &= (n-1)! 2^{\frac{(n-1)(n-2)}{2}}, \end{aligned}$$

because $f_{1,0} = 1$. This concludes the proof.

Alternatively the recurrence of $f_{n,n-1}$ can be obtained by extracting the coefficient $[z^n]$ in the following functional equation

$$T(z) = z + z^2\, T'(2z).$$

□

Lemma 4.4.4. *Both sequences (f_n) and $(f_{n,n-1})$ have the same asymptotic behaviour up to a multiplicative constant.*

Proof. Let us start with the definition of a new sequence

$$g_n = \begin{cases} 1 & \text{if } n = 1, \\ f_n / f_{n,n-1} & \text{otherwise.} \end{cases}$$

This sequence g_n satisfies the following recurrence:

$$g_n = \begin{cases} 1 & \text{if } n = 1, \\ \sum_{\ell=1}^{n-1} \sum_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i} \, 2^{\ell-i} \binom{n-\ell-1}{i-1} \, g_\ell \frac{(\ell-1)! \, 2^{(\ell-1)(\ell-2)/2}}{(n-1)! \, 2^{(n-1)(n-2)/2}} & \text{otherwise.} \end{cases}$$

When $n > 1$, extracting the term g_{n-1} from the sum we get

$$g_n = g_{n-1} + \sum_{\ell=1}^{n-2} \sum_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i} \, 2^{\ell-i} \binom{n-\ell-1}{i-1} \, g_\ell \frac{(\ell-1)! \, 2^{(\ell-1)(\ell-2)/2}}{(n-1)! \, 2^{(n-1)(n-2)/2}}.$$

Since all summands are non-negative, this implies that $g_n \geq g_{n-1}$, and thus that this sequence is non-decreasing. To prove that this sequence converges, it only remains to prove that it is (upper-)bounded.

Equation (4.27) implies that, for $n \geq 2$,

$$f_n \leq \sum_{\ell=1}^{n-1} 2^{\ell-1} \sum_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i} \binom{n-\ell-1}{i-1} f_\ell.$$

Chu-Vandermonde's identity states that, for all $\ell \leq n$,

$$\sum_{i=1}^{\min(n-\ell,\ell)} \binom{\ell}{i} \binom{n-\ell-1}{i-1} = \binom{n-1}{\ell-1}.$$

This implies the following upper-bound for f_n:

$$f_n \leq \sum_{\ell=1}^{n-1} 2^{\ell-1} \binom{n-1}{\ell-1} f_\ell = \sum_{\ell=1}^{n-1} 2^{n-\ell-1} \binom{n-1}{\ell} f_{n-\ell}$$

Using the same argument for g_n we get

$$g_n \leq g_{n-1} + \sum_{\ell=2}^{n-1} \frac{2^{(\ell-1)(\ell-2n+2)/2}}{\ell!} \, g_{n-\ell}.$$

We look at the exponent of 1 in the sum: For all $\ell \geq 2$ (as in the sum), we have $2\ell \geq \ell+2$, and thus $2n-\ell-2 \geq 2(n-\ell)$. This implies that for all $\ell \geq 2$, $(\ell-1)(\ell-2n+2)/2 \leq -(n-\ell)$, and thus that

$$g_n \leq g_{n-1} + \sum_{\ell=2}^{n-1} \frac{1}{\ell! \, 2^{n-\ell}} \, g_{n-\ell}.$$

Since the sequence $(g_n)_n$ is non-decreasing, we obtain

$$g_n \leq g_{n-1} + \frac{g_{n-1}}{2^n} \sum_{\ell=2}^{n-1} \frac{2^\ell}{\ell!} \leq g_{n-1} + g_{n-1} \frac{\mathrm{e}^2 - 3}{2^n}.$$

We set $\alpha = \mathrm{e}^2 - 3$. Iterating the last inequality, we get that

$$g_n \leq g_{n-1} \left(1 + \frac{\alpha}{2^n}\right) \leq g_1 \prod_{i=2}^{n} \left(1 + \frac{\alpha}{2^i}\right) = \exp\left(\sum_{i=2}^{n} \ln\left(1 + \frac{\alpha}{2^i}\right)\right),$$

because $g_1 = 1$. Note that, when $i \to +\infty$, we have $\ln(1+\alpha 2^{-i}) \leq \alpha 2^{-i}$ (because $\ln(1+x) \leq x$ for all $x \geq 0$). This implies that, for all $n \geq 1$,

$$g_n \leq \exp\left(\alpha \sum_{i=2}^{\infty} 2^{-i}\right) = \exp(\alpha/2).$$

In other words, the sequence $(g_n)_n$ is bounded. Since it is also non-decreasing, it converges to a finite limit c, which is also non-zero since $g_n \geq g_1 \neq 0$ for all $n \geq 1$. This is equivalent to $f_n \sim c f_{n,n-1}$ when $n \to +\infty$ as claimed. To get a lower wound on c, note that, for all $n \geq 1$, $c \geq g_n \geq g_9 = f_9/f_{9,8} \approx 1.4956$. □

Proof of Theorem 4.4.2. The latter Lemma 4.4.4 gives a proof of Theorem 4.4.2. But in order to get a better upper bound for the constant c, let us introduce another proof. In the proof of Lemma 4.4.4 we have proved

$$g_n \leq g_{n-1} + g_{n-1}\frac{\mathrm{e}^2 - 3}{2^n}.$$

We set $\alpha = \mathrm{e}^2 - 3$ and define two other sequences as

$$\bar{g}_n = \begin{cases} 1 & \text{if } n = 1 \text{ or } n = 2, \\ \bar{g}_{n-1} + \frac{\alpha}{2^n}\,\bar{g}_{n-2} & \text{otherwise,} \end{cases}$$

and

$$\bar{\bar{g}}_n = \begin{cases} 1 & \text{if } n = 1 \text{ or } n = 2, \\ \bar{\bar{g}}_{n-1} + \frac{1}{n(n+1)}\,\bar{\bar{g}}_{n-2} & \text{otherwise.} \end{cases}$$

Due to the two first terms and the recurrence equation we have for all positive n, $g_n \leq \bar{g}_n \leq \bar{\bar{g}}_n$. By induction we prove a new expression for $\bar{\bar{g}}_n$:

$$\bar{\bar{g}}_n = \begin{cases} \bar{g}_n & \text{if } n \leq 3, \\ \bar{\bar{g}}_{n-1} + \frac{2}{(n+1)!}\,a_{n-1} & \text{otherwise,} \end{cases}$$

with the sequence $(a_n)_n$ such that $a_1 = 0, a_2 = 1$ and for $n \geq 3$, $a_n = n a_{n-1} + a_{n-2}$. This sequence (a_n) is a shifted version of EIS A058307. We can either follow the work of Janson [**Jan10**] to study it, but we need less details than him so we describe here an easier approach. We define a new sequence as $b_n = a_n/n!$. We easily prove that $b_n = b_{n-1} + b_{n-2}/(n(n-1))$ with $b_1 = 0$ and $b_2 = 1/2$. Using the later recurrence, we obtain an equation satisfied by its generating function $B(z) = \sum_{n>0} b_n z^n$:

$$B(z) = \frac{z^2}{2} + zB(z) + \int_{t=0}^{u}\int_{t=0}^{z} B(u)\mathrm{d}u.$$

we thus obtain

$$(z-1)B''(z) + 2B'(z) + B(z) + 1 = 0,$$

with $B(0) = 0$ and $B'(0) = 0$. By dividing the equation by $\mathrm{i}\sqrt{1-z}$ and then by a change of variable: $u := 2\mathrm{i}\sqrt{1-z}$, we recognise the classical differential equation satisfied by Bessel functions [**BO99**]. We thus derive

$$B(z) = -1 + \frac{1}{\sqrt{1-z}}\left(\alpha\, J_1(2\mathrm{i}\sqrt{1-z}) + \beta\, Y_1(2\mathrm{i}\sqrt{1-z})\right),$$

where $J_\cdot(\cdot)$ and $Y_\cdot(\cdot)$ are the Bessel functions and α and β are two complex constants determined with the initial conditions:

$$\alpha = \frac{Y_1(2\mathrm{i}) - \mathrm{i}Y_0(2\mathrm{i})}{J_1(2)Y_0(2\mathrm{i}) + \mathrm{i}J_0(2)Y_1(2\mathrm{i})}, \qquad \beta = -\frac{J_1(2) - \mathrm{i}J_0(2)}{J_1(2)Y_0(2\mathrm{i}) + \mathrm{i}J_0(2)Y_1(2\mathrm{i})}.$$

We are interested in the asymptotic behaviour of b_n. The dominant singularity of $B(z)$ is at $z = 1$ and there

$$B(z) \underset{z\to 1}{\sim} -\frac{\beta}{\mathrm{i}\pi}\,\frac{1}{1-z}.$$

We thus deduce that b_n tends to $-\beta/(\mathrm{i}\pi) \approx 0.68894$. Since the sequence $\bar{\bar{g}}_n$ satisfies $\bar{\bar{g}}_n = \bar{\bar{g}}_{n-1} + \frac{2}{n(n+1)}\,b_{n-1}$. We deduce that the increasing sequence $(\bar{\bar{g}}_n)$ admits a finite limit. Hence it is also the case for the increasing sequence (g_n). Finally, Proposition 4.4.3 allows to conclude for the existence of the constant c. Furthermore we get

$$c < \bar{\bar{g}}_3 + \sum_{\ell\geq 4} \frac{2}{\ell(\ell+1)} \cdot \lim_{n\to\infty} b_n \approx 1.8932. \qquad \square$$

This result means that asymptotically a constant fraction of the strict monotonic general trees of size n are built in $(n-1)$ steps. For these trees, at each step of construction only one single leaf expands into a binary node. All other leaves either become a unary node or stay unchanged, meaning that on average half of the leaves will expand into unary node with one leaf expanding into a binary node. The number of internal nodes of these trees then grow like $n^2/4$.

4.4.4 Correspondence with labelled graphs

In Section 4.4.3 we defined $f_{n,k}$ the number of strict monotonic general trees of size n that have k distinct node-labels then we have shown that, for all $n \geq 1$,

$$f_{n,n-1} = (n-1)!\, 2^{\frac{(n-1)(n-2)}{2}}.$$

The factor $2^{(n-1)(n-2)/2} = 2^{\binom{n-1}{2}}$ in graphs of $(n-1)$ vertices counts the different combinations of edges (not directed) between vertices. The factor $(n-1)!$ accounts for all possible permutations of vertices. We will denote $\mathcal{S}_n$ to be the trees that $f_{n,n-1}$ counts and exhibit a bijection between strict monotonic general trees of $\mathcal{S} = \cup_{n\geq 1}\mathcal{S}_n$ with a class of labelled graphs with $n-1$ vertices defined in the following. Let us define the subclass of strict monotonic general trees $\mathcal{S} = \cup_{n\geq 1}\mathcal{S}_n$.

For all $n \geq 1$, we denote by $\mathcal{G}_n$ the set of all labelled graphs (V, ℓ, E) such that $V = \{1, \dots, n\}$, $E \subseteq \{\{i, j\} : i \neq j \in V\}$ and $\ell = (\ell_1, \dots, \ell_n)$ is a permutation of V (see Figure 4.18 for an example). We set $\mathcal{G} = \cup_{n=0}^{\infty}\mathcal{G}_n$. Choosing a graph in $\mathcal{G}_n$ is equivalent to (1) choosing ℓ (there are $n!$ choices) and (2) for each of the $\binom{n}{2}$ possible edges, choose whether it belongs to E or not (there are $2^{\binom{n}{2}}$ choices in total). In total, we thus get that $|\mathcal{G}_n| = n!\, 2^{\binom{n}{2}}$.

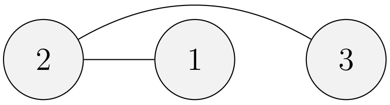


Figure 4.18: The graph $\mathcal{G}_3$ graph. In this representation, the vertices $V = \{1, \ldots, n\}$ are drawn from left to right (node 1 is the leftmost, node n is the rightmost), and their label is their image by ℓ: in this example $\ell = (2, 1, 3)$.

We recall the definitions used in Section 3.1. A size-n permutation σ is denoted by $(\sigma_1, \ldots, \sigma_n)$, and σ_i is its i-th element (the image of i), while $\sigma^{-1}(k)$ is the preimage of k (the position of k in the permutation).

Another important bijection that we will use is the bijection between binary increasing trees and permutations, see [**FS09**, page143].

We define $\mathcal{M}'' : \mathcal{S} \to \mathcal{G}$ recursively on the size of the tree it takes as an input: first, if t is the tree of size 1 (which contains only one leaf) or the tree of size 2 (one internal node attached to two leaves), then we set $\mathcal{M}''(t)$ to be the graph $(\{1\}, (1), \varnothing)$ (the graph with one vertex labelled 1 and no edge). Now assume we have defined $\mathcal{M}''$ on $\cup_{\ell=1}^{n-1} \mathcal{S}_\ell$, and consider a tree $t \in \mathcal{S}_n$. By Definition 4.4.1 and since $t \in \mathcal{S}_n$, then there exists a unique binary node in t labelled by $n-1$, and this node is attached to two leaves. Consider $\hat{t}$ the tree obtained when removing all internal nodes labelled by $n-1$ (and all the leaves attached to them) from t and replacing them by leaves. Denote by v_n the position (in, e.g., depth-first order) of the leaf of $\hat{t}$ that previously contained the binary node labelled by $n-1$ in t. Denote by $u_1, \ldots, u_m$ the positions or the leaves of $\hat{t}$ that previously contained unary nodes labelled by $n-1$ in t. We set $\mathcal{M}''(\hat{t}) = (\{1, \ldots, n-1\}, \hat{\ell}, \hat{E})$ and define $\mathcal{M}''(t) = (\{1, \ldots, n\}, \ell, E)$ where

$$\ell_i = \begin{cases} n & \text{if } i = v_n \\ \hat{\ell}_i & \text{if } \hat{\ell}_i < v_n \\ \hat{\ell}_{i-1} & \text{if } \hat{\ell}_i \geq v_n, \end{cases}$$

$E = \hat{E} \cup \{\{\hat{\ell}^{-1}(u_j), n\} : 1 \leq j \leq m\}$. An example of the bijection is depicted in Figure 4.19.

Theorem 4.4.5. *The mapping $\mathcal{M}''$ is bijective, and $\mathcal{M}''(\mathcal{S}_n) = \mathcal{G}_{n-1}$.*

Proof. From the definition, it is clear that two different trees have two distinct images by $\mathcal{M}''$, thus implying that $\mathcal{M}''$ is injective; this is enough to conclude since $|\mathcal{G}_{n-1}| = |\mathcal{S}_n|$ (see Proposition 4.4.3 for the cardinality of $\mathcal{S}_n$). □

*R*emark: It is interesting to note that this graph model is a labelled version of the binomial random graph $\mathcal{G}_n(1/2) = (V, E)$ defined as follows: $V = \{1, \ldots, n\}$ and each edge belong to E with probability $1/2$, independently from the other edges. This model, also called the Erdös-Renyi random graph was originally introduced by Erdös and Renyi [**ER59**], and simultaneously by Gilbert [**Gil59**], and has been since then extensively studied in the probability and combinatorics literature (see, for example, the books [**Bol01**] and [**Dur06**] for introductory surveys).

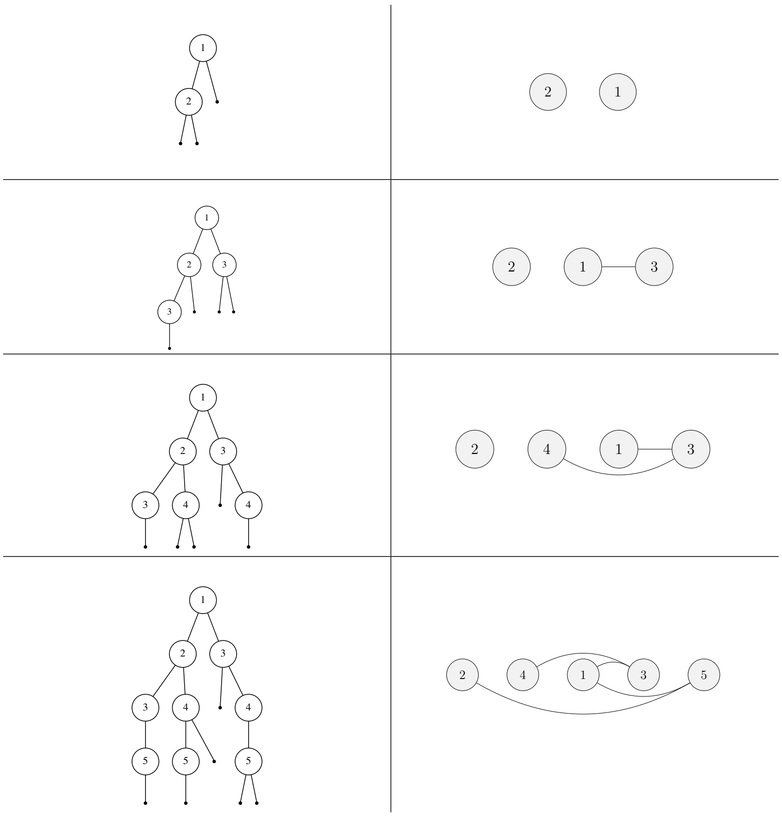


Figure 4.19: Bijection between an evolving tree in $\mathcal{S}$ from size 3 to 5 and its corresponding graph $\mathcal{G}$.

4.4.5 Analysis of typical parameters

4.4.5.1 Quantitative analysis of the number of internal nodes

Theorem 4.4.6. *Let $I_n^{\mathcal{F}}$ be the number of internal nodes in a tree taken uniformly at random among all strict monotonic general trees of size n. Then for all $n \geq 1$, we have*

$$\frac{(n-1)(n+2)}{6} \leq \mathbb{E}[I_n^{\mathcal{F}}] \leq \frac{(n-1)n}{2}.$$

To prove this theorem, we use the following proposition.

Proposition 4.4.7. *Let us denote by $s_{n,k}$ the number of strict monotonic general trees of size n that have $n-1$ distinct node-labels and k internal nodes. For all $n \geq 1$ and $k \geq 0$,*

$$s_{n,k} = (n-1)!\binom{(n-1)(n-2)/2}{k-(n-1)},$$

and thus, if $I_n^{\mathcal{S}}$ is the number of internal nodes in a tree taken uniformly at random among all strict monotonic general trees of size n that have $n-1$ distinct label nodes, then, for all

$n \geq 1$,

$$\mathbb{E}[I_n^{\mathcal{S}}] = \frac{(n-1)(n+2)}{4}.$$

Proof. Let us prove the formula for $s_{n,k}$ by induction. For $n = 1$, k can only be 0 thus $s_{1,0} = 1 = 0!\binom{0}{0}$.

We suppose that $s_{m,k} = (m-1)!\binom{(m-1)(m-2)/2}{k-(m-1)}$ holds for $m = n-1$ and $k \in \{n-1, \ldots, (n-2)(n-3)/2\}$.

Then, we are interested in the value of $s_{n,k}$:

$$s_{n,k} = \sum_{s=0}^{k-(n-1)} (n-2)!\binom{(n-2)(n-3)/2}{s-(n-2)}\binom{n-1}{k-s-1}(k-s-1).$$

Let $k' = k-(n-1)$ and $s' = s-(n-2)$. Replacing k' and s' in the equation gives,

$$\begin{aligned}\tilde{s}_{n,k'} &= \sum_{s'=0}^{k'}(n-2)!\binom{(n-2)(n-3)/2}{s'}\binom{n-1}{k'-s'+1}(k'-s'+1)\\ &= (n-1)!\sum_{s'=0}^{k'}\binom{(n-2)(n-3)/2}{s'}\binom{n-2}{k'-s'}.\end{aligned}$$

Using Chu-Vandermonde identity, we finally obtain

$$s_{n,k} = (n-1)!\binom{(n-1)(n-2)/2}{k-(n-1)}.$$

We now can compute the average number of internal nodes of $\mathcal{S}_n$:

$$\mathbb{E}_n[I_n^{\mathcal{S}}] = \frac{\sum_{k=n-1}^{n(n-1)/2} k(n-1)!\binom{(n-1)(n-2)/2}{k-(n-1)}}{(n-1)!\,2^{(n-1)(n-2)/2}}.$$

Again we reverse the sum: $k' = k-(n-1)$,

$$\begin{aligned}\mathbb{E}[I_n^{\mathcal{S}}] &= \frac{\sum_{k'=0}^{(n-1)(n-2)/2} (k'+(n-1))(n-1)!\binom{(n-1)(n-2)/2}{k'}}{(n-1)!\,2^{(n-1)(n-2)/2}}\\ &= \frac{\sum_{k'=0}^{(n-1)(n-2)/2} k\,'\binom{(n-1)(n-2)/2}{k'} + (n-1)\sum_{k'=0}^{(n-1)(n-2)/2}\binom{(n-1)(n-2)/2}{k'}}{2^{(n-1)(n-2)/2}}\\ &= \frac{(n-1)(n-2)}{4} + (n-1) = \frac{(n-1)(n+2)}{4}.\end{aligned}$$

□

We are now ready to prove the main theorem of this section.

Proof of Theorem 4.4.6. Note that the number of internal nodes of a strict monotonic general tree of size n belongs to $\{1, \dots, n(n-1)/2\}$. The upper bound follows from the fact that, at the ℓ-th iteration in Definition 4.4.1, a maximum of ℓ internal nodes is added to the tree, and $\sum_{\ell=1}^{n} \ell = n(n-1)/2$. In particular, we thus have that, almost surely for all $n \geq 1$, $I_n^{\mathcal{F}} \leq n(n-1)/2$, and thus $\mathbb{E}[I_n^{\mathcal{F}}] = \mathcal{O}(n^2)$.

For the lower bound, we denote by $\mathcal{S}_n$ the set of strict monotonic general trees of size n that have $n-1$ distinct node-labels. Moreover, we denote by t_n a tree taken uniformly at random in $\mathcal{F}_n$, and by $I_n^{\mathcal{F}}$ its number of internal nodes. We have, for all $n \geq 1$,

$$\begin{aligned}\mathbb{E}[I_n^{\mathcal{F}}] &= \mathbb{E}[I_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] \cdot \mathbb{P}(t_n \in \mathcal{S}_n) + \mathbb{E}[I_n^{\mathcal{F}} \mid t_n \notin \mathcal{S}_n] \cdot \mathbb{P}(t_n \notin \mathcal{S}_n) \\ &\geq \mathbb{E}[I_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] \cdot \mathbb{P}(t_n \in \mathcal{S}_n) = \mathbb{E}[I_n^{\mathcal{S}}] \cdot \frac{f_{n,n-1}}{f_n},\end{aligned}$$

where we have used conditional expectations and the fact that conditionally on being in $\mathcal{S}_n$, t_n is uniformly distributed in this set, and, in particular, $\mathbb{E}[I_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] = \mathbb{E}I_n^{\mathcal{S}}$. Using Proposition 4.4.7 and the upper bound of Proposition 4.4.3, we thus get

$$\mathbb{E}[I_n^{\mathcal{F}}] \geq \frac{2}{3}\,\frac{(n-1)(n+2)}{4},$$

which concludes the proof. □

4.4.5.2 Quantitative analysis of the number of distinct labels

Theorem 4.4.8. *Let $X_n^{\mathcal{F}}$ denotes the number of distinct internal-node labels (or construction steps) is a tree taken uniformly at random among all strict monotonic general trees of size n, then for all $n \geq 1$,*

$$\frac{2}{3}\,(n-1) \leq \mathbb{E}[X_n^{\mathcal{F}}] \leq n-1.$$

Proof. First note that since at every construction step in Definition 4.4.1 we add at least one leaf in the tree, then after ℓ construction steps, there are exactly ℓ distinct labels and at least $\ell+1$ leaves in the tree. Therefore, $n \geq X_n^{\mathcal{F}} + 1$ almost surely for all $n \geq 1$, which implies in particular that $\mathbb{E}[X_n] \leq n-1$, as claimed.

For the lower bound, we reason as in the proof of Theorem 4.4.6, and using the same notations:

$$\mathbb{E}[X_n^{\mathcal{F}}] \geq \mathbb{E}[X_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] \cdot \mathbb{P}(t_n \in \mathcal{S}_n) = (n-1)\,\frac{f_{n,n-1}}{f_n},$$

because $\mathbb{E}[X_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] = n-1$ by definition of $\mathcal{S}_n$ (being the set of all strict monotonic general trees of size n that have $n-1$ distinct node-labels). Using the upper bound of Proposition 4.4.3 gives that $\mathbb{E}[X_n^{\mathcal{F}}] \geq 2(n-1)/3$, which concludes the proof. □

4.4.5.3 Quantitative analysis of the height of the trees

Theorem 4.4.9. *Let $H_n^{\mathcal{F}}$ denotes the height of a tree taken uniformly at random in $\mathcal{F}_n$, the set of all strict monotonic general trees of size n. Then we have, for all $n \geq 0$,*

$$\frac{n}{3} \leq \mathbb{E}[H_n^{\mathcal{F}}] \leq n-1.$$

To prove this theorem, we first prove the following:

Proposition 4.4.10. *Let us denote by $H_n^{\mathcal{S}}$ the height of a tree taken uniformly at random in $\mathcal{S}_n$, the set of all strict monotonic general trees of size n that have $n-1$ distinct labels. Then we have, for all $n \geq 0$,*

$$\frac{n}{2} \leq \mathbb{E}[H_n^{\mathcal{S}}] \leq n-1.$$

Proof. Define the sequence of random trees $(t_n)_{n\geq 0}$ recursively as:
- t_1 is a single leaf.
- Given t_{n-1}, we define t_n as the tree obtained by choosing a leaf uniformly at random among all leaves of t_{n-1}, replacing it by an internal nodes to which two leaves are attached, and, for each of the other leaves of t_{n-1}, choose with probability $1/2$ (independently from the rest) whether to leave it unchanged or to replace it by a unary node to which one leaf is attached.

One can prove by induction on n that for all $n \geq 1$, t_n is uniformly distributed in $\mathcal{S}_n$. We denote by $H_n^{\mathcal{F}}$ the height of t_n. Since the height of t_n is at most the height of t_{n-1} plus 1 for all $n \geq 2$, we get that $H_n^{\mathcal{S}} \leq n-1$ almost surely.

For the upper bound, we note that, for the height of t_n to be larger than the height of t_{n-1}, we need to have replaced at least one of the maximal-height leaves in t_{n-1}. There is at least one leaf of t_{n-1} which is at height $H_{n-1}^{\mathcal{S}}$ and this leaf is replaced by an internal node with probability

$$\frac{1}{2}\left(1-\frac{1}{n-1}\right)+\frac{1}{n-1} \geq \frac{1}{2}.$$

Therefore, for all $n \geq 1$, we have

$$\mathbb{P}(H_n^{\mathcal{S}} = H_{n-1}^{\mathcal{S}}+1) \geq \frac{1}{2},$$

which implies, since $H_n^{\mathcal{S}} \in \{H_{n-1}^{\mathcal{S}}, H_{n-1}^{\mathcal{S}}+1\}$ almost surely,

$$\mathbb{E}[H_n^{\mathcal{S}}] = \mathbb{E}[H_{n-1}^{\mathcal{S}}] + \mathbb{P}(H_n^{\mathcal{S}} = H_{n-1}^{\mathcal{S}}+1) \geq \mathbb{E}[H_{n-1}^{\mathcal{S}}] + \frac{1}{2}.$$

Therefore, for all $n \geq 1$, we have $\mathbb{E}[H_n^{\mathcal{S}}] \geq \mathbb{E}[H_0^{\mathcal{S}}] + n/2 = n/2$, as claimed. □

Proof of Theorem 4.4.9. By Definition 4.4.1, it is straightforward to see that the height of a tree built in ℓ steps is at most ℓ since the height increases by at most one per construction step. Since a tree of size n is built in at most $n-1$ steps, we get that $H_n^{\mathcal{F}} \leq n-1$ almost surely, which implies, in particular, that $\mathbb{E}[H_n^{\mathcal{F}}] \leq n-1$.

For the lower bound, note that, if t_n is a tree taken uniformly at random in $\mathcal{F}_n$ and $H_n^{\mathcal{F}}$ is its height, then

$$\mathbb{E}[H_n^{\mathcal{F}}] \geq \mathbb{E}[H_n^{\mathcal{F}}|t_n \in \mathcal{S}_n] \cdot \mathbb{P}(X \in \mathcal{S}_n) \geq \frac{2}{3}\,\mathbb{E}[H_n^{\mathcal{S}}],$$

where we have used Proposition 4.4.3 and the fact that t_n conditioned on being in $\mathcal{S}_n$ is uniformly distributed in this set and thus $E[H_n^{\mathcal{F}} \mid t_n \in \mathcal{S}_n] = \mathbb{E}H_n^{\mathcal{S}}$. By Proposition 4.4.10, we thus get $\mathbb{E}[H_n^{\mathcal{F}}] \geq {}^{n}/_{3}$, as claimed. □

4.4.5.4 Quantitative analysis of the depth of the leftmost leaf

Theorem 4.4.11. *Let us denote by $D_n^{\mathcal{F}}$ the height of a tree taken uniformly at random in $\mathcal{F}_n$, the set of all strict monotonic general trees of size n. Then we have, for all $n \geq 0$,*

$$\frac{n}{3} \leq \mathbb{E}[H_n^{\mathcal{F}}] \leq n-1.$$

Proposition 4.4.12. *Let us denote by $D_n^{\mathcal{S}}$ the depth of the leftmost leaf of a tree taken uniformly at random in $\mathcal{S}_n$, the set of all strict monotonic general trees of size n that have $n-1$ distinct labels. Then we have, for all $n \geq 0$,*

$$\frac{n}{2} \leq \mathbb{E}[D_n^{\mathcal{S}}] \leq n-1.$$

Proof. Given the uniform process of trees t_n presented in Proposition 4.4.10. The depth of the leftmost leaf is always smaller than $n-1$. Let X_n be a Bernoulli variable taking value 1 if the leftmost leaf of t_n has been expanded at iteration n and the value 0 otherwise. Then for $n \geq 1$,

$$\mathbb{P}\left(X_n = 1\right) = \frac{1}{n} + \frac{(n-1)}{n}\frac{1}{2} = \frac{n+1}{2n} \geq \frac{1}{2}.$$

Since at each iteration step either the leftmost leaf expand to make a binary node which gives $\frac{1}{n}$ or it has not created a binary and then it has $\frac{1}{2}$ probability to make a unary node. The depth of the leftmost leaf is $D_n^{\mathcal{S}} = \sum_{k=1}^{n} X_k$. Therefore for $n \geq 1$,

$$\mathbb{E}[D_n^{\mathcal{S}}] \geq \frac{n}{2}.$$

Which concludes the proof. □

Proof of Theorem 4.4.11. By the same arguments as in Theorem 4.4.9 the result follows directly since we have the same bounds on the depth of leftmost leaf as we had in the height of the tree. □

4.5 Conclusion

As a conclusion, we comment our main analytical results (summarised in Table 4.1, Table 4.7 and Table 4.2) in the light of the simulations obtained using the different random samplers designed in Chapter 7 (see the right-hand sides of Figure 4.2, Figure 4.9 and Figure 4.16), and compare on the similarities and difference of our three models. Recall that in the representations no label is represented but the length of an edge between two internal nodes is proportional to the difference of the labels of the nodes it connects.

A few of our analytical results can be observed looking at the simulations in Chapter 7: for example, the fact that a large proportion of the nodes are binary in a large monotonic Schröder tree, which we have confirmed by a rigorous analysis (see Theorem 4.2.24), is visible on Figure 4.2. From Figure 4.9, one could conjecture this is also true in the case of strict monotonic Schröder trees, but this question remains open.

From Figure 4.2, Figure 4.9 and Figure 4.16 it seems clear that the model of *strict monotonic general Schröder* trees behaves drastically differently from the two other models, which are quite similar. This is indeed what we have proved in our analysis: for example, the height of a typical strict monotonic general tree of size n is of order $\Theta(n)$ (see Theorem 4.4.9), while we have shown that a in the monotonic case, the height is of order $\Theta(\log n)$ (see Theorem 4.2.27). Another huge difference is that the number of internal nodes in a large typical monotonic general Schröder tree is of order $\Theta(n^2)$ (see Theorem 4.4.6) while, in the two other models, this parameter is of order n (see Theorem 4.2.18 Theorem 4.3.10).

Proving results on the height of different families of random trees is often a challenging question, and we have seen that it is indeed one of the most intricate parameters to study in our three models: in the case of monotonic and strict monotonic general trees, we obtain a Θ-estimate but we only obtain a $\ln n$ lower bound in the case of strict monotonic Schröder trees (see Section 4.3.5.4). A natural conjecture, based on the fact that *increasing Schröder* and *strict monotonic Schröder* trees seem to behave similarly in some aspects, we conjecture that the height of a typical strict monotonic Schröder tree is also of order $c \ln n$, for some constant c.

CHAPTER 5

General asymptotics for varieties of monotonic Schröder trees

Contents

> أَلَمْ تَرَ كَيْفَ ضَرَبَ اللَّهُ مَثَلًا كَلِمَةً طَيِّبَةً
> كَشَجَرَةٍ طَيِّبَةٍ أَصْلُهَا ثَابِتٌ وَفَرْعُهَا فِي
> السَّمَاءِ[1].
>
> *Quran 14:24*

[1] Have you not considered how God presents an example, [making] a good word like a good tree, whose root is firmly fixed and its branches [high] in the sky?

5.1 Introduction

In this chapter we study of an *evolution process* that generalises the processes studied in Chapter 4 and incorporates them as special cases. The idea is to study evolution processes, where individuals evolve at some period of times and give birth to new individuals and the process goes on. The process that we present is general in the sense that it can parameterised by two variables. One variable controls the number of children that an individual can reproduce at each period of time. The process is general enough to incorporate also the fact that the number of newly born children from an individual can be unbounded. The second variable, is not a local variable but a global one, and this controls how many individuals can reproduce at each period of time.

The result of the *evolution process* can be represented with a plane rooted tree that has labels on internal nodes to denote the period of times in which this individual appeared. Leaves (i.e external nodes) are not labelled and represent the individuals that can reproduce and transform to internal nodes with new leaves.

Evolution processes are known in different fields. In combinatorics, several models of evolution processes have been studied. Many classes of trees in combinatorics are described by an evolution process like *simple trees* [**MM78**], *Schröder trees* and *phylogenetic trees* [**Sch70**], *increasing trees* [**BFS92**], *bifurcating trees* and *multifurcating trees* [**CPQ96**]. An introduction into various aspects of trees in combinatorics can be found [**Drm09**] and for trees arising in phylogenetics [**Fel03**]. See Section 3.5.1 for more details on the tree classes cited above.

In probability theory, evolution processes, are described by rules of individual reproduction according to some probability distribution. Some famous models include Galton-Watson trees formalised by Neveu in [**NEV86**], Yule trees introduced by Pittel in [**Pit84**]. For some tree classes, there exist a combinatorial evolution process and a probabilistic evolution process that generates uniformly a tree from all trees of the same size. But it is not always the case.

Our main contributions in this chapter concerns asymptotic enumeration of different classes of trees that can be defined through the evolution process. The type of theorem we look for is close to the one on the asymptotics of *simple trees* that we have presented in Theorem 3.5.15. This theorem uses a *characteristic equation* and the main asymptotic term comprises derivatives of the degree function $\phi(u)$. We look to develop the same kind of theorem for our *evolution process*. We will also find some new results on the asymptotics of labelled binary trees under different models of increasing labellings, see Table 5.8. For instance we will have the asymptotics of binary trees with weakly increasing labellings along branches. We will also be able to enumerate d-ary trees with weakly increasing labellings along branches. Therefore, this study constitutes as far as we know a first account on weakly increasing labellings in the sense that we define, with the root always having label 1, the labels are weakly increasing along branches and finally there are no holes in the labels (if k is the maximum label then all labels between 1 and k must appear).

As we will see the specification of the *evolution process* gives rise invariably to *divergent generating functions*. The key idea in the proofs of our theorems is related to the use of *Borel* *transform* of the recurrence of the coefficients. In combinatorics the *Borel transform* plays a role similar to *Borel summation* for divergent power series.

We also see in this chapter how using specifications belonging to the unlabelled world of the symbolic method can help us define different increasing labellings on tree structures.

We will start this chapter in Section 5.2, by giving the formal definition of the evolution process. From the formal definition we will go on to give the statements of the main two theorems to be proved (Theorem 5.2.6 and Theorem 5.2.7). Our results in this chapter are mainly results relating to the enumeration of tree classes and general asymptotic equivalent results. Then a section is dedicated for applications of the theorems on specific tree classes in Section 5.3. In the section that follows, we will write a combinatorial recurrence on the coefficients of this process in Section 5.4. The four following sections (Section 5.5,Section 5.6,Section 5.7 and Section 5.8) are dedicated to the proofs of the main results. Our results require that binary nodes are allowed to appear in the resulting tree. This is why in Section 5.10, we summarise some of the results where this is not the case and give some open questions about the evolution process in this case. We end this chapter with a conclusion that summarises the results in Section 5.11.

5.2 Formal definition and main results

We present now the evolution process that we will study in this chapter. This process generates trees such that their labels along branches are strictly increasing. But it will be possible to relax the strict increasing condition when we present a variant of this process later on.

Definition 5.2.1. ***(Evolution process for a variety of strict monotonic Schröder trees)*** *Given some colored degree function $\phi(z)$ as defined in Definition 3.5.6 and $r \subset \mathbb{N}^*$, the following evolution process generates a variety of strict monotonic Schröder tree parameterised by $\phi(z)$ and r. Let $min(r) = 1$[2]. The process starts at time-step 0 with a single leaf and at each time-step $i \geq 1$ is as follows:*

(1) Choose a nonempty subset of leaves L such that $|L| \in r$.
(2) For each leaf $\ell \in L$ choose an admissible degree and a colour (d, c), meaning that $\phi_d > 0$ and $1 \leq c \leq \phi_d$. There also exists some leaf, such that the associated couple (d, c) is such that $d > 1$.
(3) Replace each leaf ℓ with an internal node labelled by i with colour c and having d new leaves attached to it.

Remark 5.2.2. *If $min(r) \neq 1$, it means that we have more than one repetition at each iteration step. Therefore, it is not possible to start with single leaf. In this case we start with a tree consisting of a root labelled 1 that have $min(r)$ leaves. Then the process goes on as described above and starts at time step 2 rather than one.*

This evolution process generates a *strict monotonic Schröder tree* since at each iteration step we add some nodes having the same label in different places of the tree such that the labelling

[2] where $min()$ returns the smallest value of a set

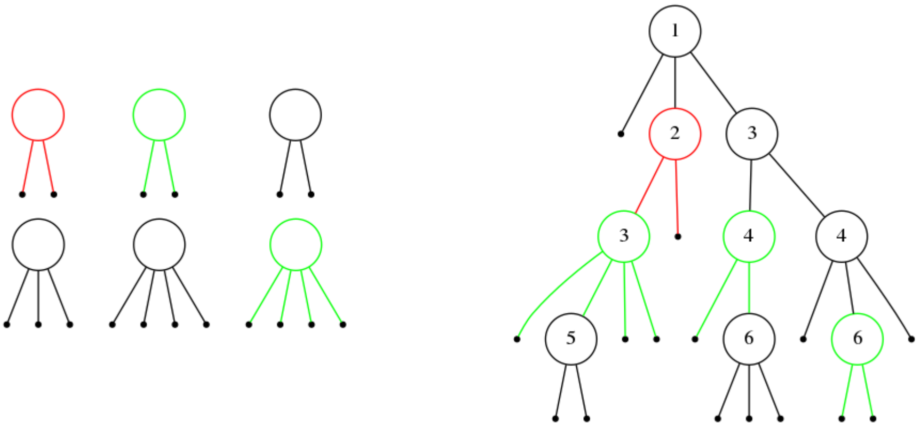


Figure 5.1: Example of Definition 5.2.1. (Left) An example of the colored degree function, here $\phi(z) = 3z^2 + z^3 + 2z^4$, there are 3 colours of binary nodes, 1 colour for ternary nodes and 2 colours for quaternary nodes. (Right) A tree of size 15 built with $\phi(z)$ where the set of allowed repetitions $r = \{1, 2, 3\}$, so that each label can appear 1,2 or 3 times. But they might not all appear, in the example there is no node with 3 repetitions.

along each branch is always strictly increasing. In Section 5.3 we will see how to extend this process in order to generate *monotonic trees* (where the labelling along branches is weakly increasing).

Translating the above process using the framework of the *symbolic method* (see [**FS09**]), we obtain the following functional relation for the generating series $B^{r,\phi}$ enumerating trees built via the evolution process based on the *colored degree function* $\phi(z)$, and the set of allowed repetitions $r \subset \mathbb{N}^*$ and let $m = min(r)$:

$$B(z) = z^m + \sum_{i \in r} \frac{1}{i!} B^{(i)}(z) \left(\phi(z)^i - (\phi_1\, z)^i\right). \tag{5.1}$$

Where $B^{(i)}(z)$ is the $i-th$ derivative of $B(z)$. We write B instead of $B^{r,\phi}$ for simplicity but it should be always clear that B is characterized by these two parameters.

Remark 5.2.3. *We notice that the condition "There also exists some leaf, such that the associated couple (d, c) is such that $d > 1$" in the evolution process Definition 5.2.1 is reflected in the specification by the factor $-(\phi_1 z)^i$, which means that at each iteration step at least one leaf evolves to something different from a unary node, this is why we erase the configuration where all leaves evolve to unary nodes. This ensures the resulting class to be a combinatorial class with a finite number of objects for each size.*

Table 5.1 sums up some of the most important examples that this evolution process captures. More examples are discussed in Section 5.3.

Supertrees are a class of rooted plane binary increasing trees where labels can appear twice. It has been introduced as a model of tree of life in [**SDH$^+$04**]. This class can be specified using our *evolution process* with $\phi(z) = z^2$ and $r = \{1, 2\}$. Monotonic d-ary trees will be

r	$\phi(z)$	Name	References
$\{1\}$	z^d	Plane d-ary increasing	[**BFS92**]
$\{1\}$	$\frac{z^2}{1-z}$	Increasing Schröder	Section 4.2,[**BGN19**]
$\mathbb{N}^*$	z^2	Strict monotonic binary	[**BGGW20**]
$\mathbb{N}^*$	$\frac{z^2}{1-z}$	Strict monotonic Schröder	Section 4.3,[**BGN19**]
$\mathbb{N}^*$	$\frac{z}{1-z}$	Strict monotonic general Schröder	Section 4.4,[**BGMN20**]
$\mathbb{N}^*$	plane d-ary	monotonic d-ary trees	Section 5.3,[**BGNS20**]
$\{1,2\}$	z^2	Supertrees	[**SDH$^+$04**]
$\{d\}$	z^2	Increasing binary with d label repetitions	Section 5.3.2

Table 5.1: Some of examples of tree classes covered by Definition 5.2.1

introduced in Section 5.3. They are rooted plane trees such that the labellings along branches is weakly increasing.

Conditions on the growth of $\phi(z)$

Our asymptotic results work under conditions of growth of the coefficients ϕ_i in $\phi(z)$. However, the conditions are general enough, so that, the results hold for interesting cases as will be presented in Section 5.3. Finally, it is possible that the growth condition on $\phi(z)$ can be relaxed further in some cases. But we keep it like this for uniformity of the results.

Condition 5.2.4. *Let $r \subset \mathbb{N}^*$ and $m = min(r)$. Let $\phi(z)$ be a coloured degree function as presented in Definition 3.5.6 and such that $\phi_1 = 0$, $\phi_2 \geq 1$ and $\phi_n \underset{n\to\infty}{=} O\left(\frac{n!}{m!^{n/m}\, n^{m+4}}\right)$.*

Condition 5.2.5. *Let $\phi(z)$ be a coloured degree function as presented in Definition 3.5.6 and such that $\phi_1 \geq 1$, $\phi_2 \geq 1$ and $\phi_n = O\left(\frac{n!}{n^5}\right)$.*

The main difference between both conditions Condition 5.2.4 and Condition 5.2.5 is that in the first $\phi_1 = 0$ and in the second this condition is negated so that $\phi_1 \geq 1$.

The main theorems are:

Theorem 5.2.6. *Let $\phi(z)$ be as in Condition 5.2.4, with $r \neq \emptyset$. Let $m = min(r)$, when n tends to ∞ and is of the form $n \equiv 0 mod\, m$,*

$$B_n^{r,\phi} \underset{n\to\infty}{\sim} \kappa\, n! \left(\frac{\phi_2}{\rho}\right)^n n^{-1+\frac{\rho\,\phi_3}{\phi_2^2}-\frac{\rho\, f''(\rho)}{f'(\rho)}},$$

where κ is a constant that depends on $\phi(z)$ and r. Let $f(z) = \sum\limits_{i=1, i\in r}^{\infty} \frac{z^i}{i!}$, then ρ is the positive real solution of smallest modulus of the equation $f(z) - 1 = 0$.

Theorem 5.2.7. *Let $\phi(z)$ be as in Condition 5.2.5 , let $r \subset \mathbb{N}^*$, $r \neq \emptyset$, and $r \neq \{1\}$, then when n tends to ∞,*

$$B_n^{r,\phi} \underset{n\to\infty}{\sim} \kappa\, (n-1)!\, \phi_2^{n-1} \prod_{k=1}^{n-1} \left(\sum_{i=1, i\in r}^{\infty} \phi_1^{i-1} \binom{n-k-1}{i-1} \right),$$

where κ is a constant that depends on $\phi(z)$ and r.

These two theorems show that the asymptotic first order is only affected by the by ϕ_2 and ϕ_3 if there are no unary nodes (i.e $\phi_1 = 0$). However, when unary nodes are allowed (i.e $\phi_1 > 0$) only ϕ_1 and ϕ_2 appear in the first order asymptotic. In both cases as stated in the theorems $\phi_2 > 0$.

For the case where unary nodes are not allowed, we study some relaxations of the condition on $\phi_2 \geq 1$, in Section 5.10. For the second case, where unary nodes are allowed an extension of Theorem 5.2.7 is given in Theorem 5.9.7.

When r consists of consecutive integers that starts from 1, that is $r = [1, 2, \dots, m]$ or $r = \mathbb{N}^*$, then a specialising theorem gives the following:

Theorem 5.2.8. *Let $\phi(z)$ be as in Condition 5.2.4, we have,*

$$B_n^{[m],\phi} \underset{n\to\infty}{\sim} \kappa\, n! \left(\frac{\phi_2}{\rho}\right)^n n^{-1+\tau+\rho\left(-1+\frac{\phi_3}{\phi_2^2}\right)},$$

where κ is a constant that depends on $\phi(z)$ and m, and is given by an implicit equation, let $f(z) = \sum\limits_{i=1}^{m} \frac{z^i}{i!}$, ρ is the positive real solution of smallest modulus of the equation $f(z) - 1 = 0$ and $\tau = \frac{\rho^m}{(m-1)! f'(\rho)}$. Moreover, the numbers ρ and τ are algebraic except when $r = \mathbb{N}^*$, then ρ becomes transcendental.

We will start the proofs sections by showing two specific cases of r. In Section 5.5 we give the proof for $r = \{d\}$, and in Section 5.6 we give the proof for $r = \mathbb{N}^*$.

The statement of the two cases are:

Proposition 5.2.9. *Let $\phi(z)$ be as in Condition 5.2.4, and $r = \{d\}$, with $d \geq 1$, then when n tends to ∞ and is of the form $n \equiv 0 \bmod d$,*

$$B_n^{\{d\},\phi} \sim \kappa\, n! \left(\frac{\phi_2}{d!^{n/d}}\right)^n n^{-d+\frac{\phi_3}{\phi_2^2}}.$$

Where κ is a constant that depends on ϕ and is given by an implicit equation.

Corollary 5.2.10. *Let $\phi(z)$ be as in Condition 5.2.4, and $r = \mathbb{N}^*$. Then,*

$$B_n^{\mathbb{N}^*,\phi} \underset{n\to\infty}{\sim} \kappa\, n! \left(\frac{\phi_2}{\ln 2}\right)^n n^{\left(-1+\frac{\phi_3}{{\phi_2}^2}\right)\ln 2 - 1}.$$

Where κ is a constant that depends on ϕ and is given by an implicit equation.

The result in Proposition 5.2.9 is used to make Condition 5.2.4.

Corollary 5.2.10 is put as a corollary because it follows directly from Theorem 5.2.8, but we will make a proof of it since the proof of the general result follows the same schema with a striction on the summands of the specialisation.

Then we go to show the general case Theorem 5.2.6 in Section 5.7 and its specialisation Theorem 5.2.8 in Section 5.8. Finally, we will show the proof of Theorem 5.2.7 where unary nodes are allowed in Section 5.9.

Principles of the proofs:

The proofs in the next sections follow the same steps which are summarised in the following:

- Make a *Borel transform* (rescaling by $n!$) on the coefficients of the main recurrence.
- Identify the coefficients of the leading behaviour.
- Write a differential equation that satisfies an upper and a lower bound of the *generating function*.
- Prove an upper and lower bounds on the coefficients (i.e a Θ result).
- From the Θ result and a careful analysis of the error terms deduce the equivalent result.

5.3 Applications

All applications that we will present in this section are parameterised version of the evolution process presented in Definition 5.2.1 which gives rise to the following functional equation Equation (5.1) that we recall here.

$$B(z) = z^m + \sum_{i \in r} \frac{1}{i!} B^{(i)}(z) \left(\phi(z)^i - (\phi_1 \, z)^i \right) .$$

This equation is parameterised by:

- The function $\phi(z)$ which represents the set of allowed nodes degrees with their number of colours.
- The set r which represents the set of allowed number repetitions for an each iteration step and $m = min(r)$.

All the asymptotic results presented are applications of Theorem 5.2.6 when there are no unary nodes and Theorem 5.2.7 otherwise.

We start this section by presenting a variant of the *evolution process* of Definition 5.2.1 that generates trees with weakly increasing labellings in Section 5.3.1.

In Section 5.3.2, we talk about families of *strict monotonic Schröder* trees, where there can be label repetitions on different branches of the tree but the labellings are strictly increasing from the root to any leaf. Using our theorem we derive again two known asymptotic results. We additionally study the behaviour of *increasing binary trees* with a fixed number of repetitions at each iteration step.

In Section 5.3.3 we consider trees with weakly increasing labellings. We study two varieties of them. The first one is called connected monotonic Schröder trees where the labellings are weakly increasing from the root to any leaf but where the same labels all belong to the same subtree.

The second variety that is called monotonic Schröder trees is the one of trees with weakly increasing labellings along branches where there can be label repetitions on different branches of the trees.

In Section 5.3.4, we give some examples of *labelled Schröder trees* with weakly increasing labellings, before we pass on to Section 5.3.5 where we talk about plane d-ary trees with weakly increasing labellings that we call then *monotonic d-ary Schröder trees*.

We then end with Section 5.3.6, that gives two examples of trees with unary nodes.

5.3.1 Double nature of $\phi(z)$

The *evolution process* in Definition 5.2.1 sees the ***coloured degree function*** (defined in Definition 3.5.6) representing the set of allowed arities and node colours in a tree and thus it generates a *strict monotonic Schröder tree* since at each iteration step we add some nodes having the same in different places of the tree such that the labelling is always strictly increasing as in [**BGGW20, BGN19**].

Alternatively, the coefficients ϕ_i, $i \geq 2$, can be seen as the number of trees with i leaves belonging to some class of plane rooted unlabelled trees (in the sequel, we will refer to elements of such classes as *tree-shapes*). In this second context the objects that we will construct are *monotonic trees* as defined in Definition 3.5.23.

In order to generate *monotonic trees*, in which case the coefficients of $\phi(z)$ are alternatively interpreted as enumerating tree-shapes rather than node colours, a slight modification of the evolution process Definition 5.2.1 is required: at each iteration step i, each selected leaf is replaced by a tree-shape, rather than a coloured internal node, and all internal nodes of this tree-shape are labelled by i.

Definition 5.3.1. ***(Evolution process for a variety of monotonic Schröder trees)***
Given $\phi(z)$ that represents the generating function of a rooted unlabelled plane class of tree counted by its number of leaves (we will call it the tree shapes) starting at size 2 and $r \subset \mathbb{N}^*$, the following evolution process generates a variety of monotonic Schröder tree parameterised by $\phi(z)$ and r. Let $min(r) = 1$[3]. The process starts at time-step 0 with a single leaf and at each time-step $i \geq 1$ is as follows:

(1) Choose a nonempty subset of leaves L such that $|L| \in r$.
(2) For each leaf $\ell \in L$ choose an admissible size and a tree-shape (s, t), meaning that $\phi_s > 0$ and $1 \leq t \leq \phi_s$. There also exists some leaf, such that the associated couple (s, t) is such that $s > 1$.
(3) Replace each leaf ℓ with its corresponding the tree shape t having all its internal nodes labelled i.

An example of this evolution process is presented in Figure 5.2. We now exhibit examples of asymptotic enumeration for a number of interesting combinatorial tree classes as direct applications of the results presented in Section 5.2.

[3] where $min()$ returns the smallest value of a set

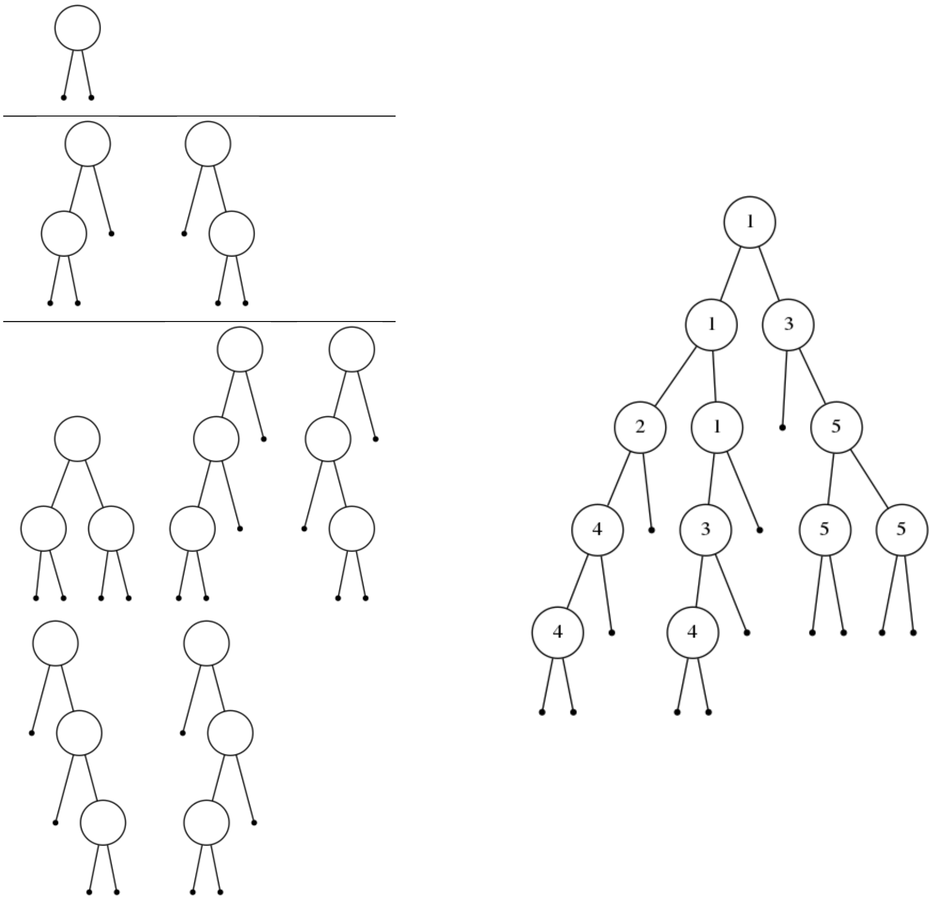


Figure 5.2: Example of Definition 5.3.1. (Left) An example of the colored degree function, seen as a class of rooted unlabelled trees here binary trees $\phi(z) = z^2 + 2z^3 + 5z^4$, the trees of first sizes $2, 3$ and 4 are depicted. At each iteration step a leaf can evolve into a whole unlabelled shape and that all its internal nodes get the label of the current iteration step. (Right) A resulting monotonic binary tree, here $r = \mathbb{N}^*$.

	r	$\phi(z)$	Definition	Application
Varieties of strict monotonic Schröder trees	r	Coloured degree function	Definition 3.5.24	Section 5.3.2
Varieties of connected monotonic Schröder trees	$\{1\}$	Tree shapes	Definition 3.5.23	Section 5.3.3
Varieties of monotonic Schröder trees	r	Tree shapes	Definition 3.5.26	Section 5.3.4

Table 5.2: Increasing labellings on Schröder trees generated by Definition 5.2.1 and Definition 5.3.1.

5.3.2 Varieties of strict monotonic Schröder trees

We recall the reader that we defined strict monotonic Schröder trees in Definition 3.5.24 of Section 3.5.6. Labelled Schröder trees are plane rooted trees, counted by their number of leaves. Only internal nodes bear labels. Then a variety of *strict monotonic Schröder trees* are labelled Schröder trees, such that the labelling is strictly increasing from the root to any leaf and if k is the maximum label in the tree, all labels from 1 to k appear. In particular, k can be smaller than the number of internal nodes and therefore, we can have label repetitions.

In these classes, the Definition 5.2.1 is such that $r = \mathbb{N}^*$ and $\phi_1 = 0$ (when unary nodes are allowed we added the word general in the name, see Section 5.3.6). Therefore, the sum in Equation (5.1) can be expressed in terms of composition of functions (substitution):

$$B(z) = z + B(\phi(z) + z) - B(z).$$

In the substitution $B(\phi(z) + z)$ we add z which then allows for each leaf to be expanded or not, so that any subset of leaves can evolve which is also represented by the unbounded sum.

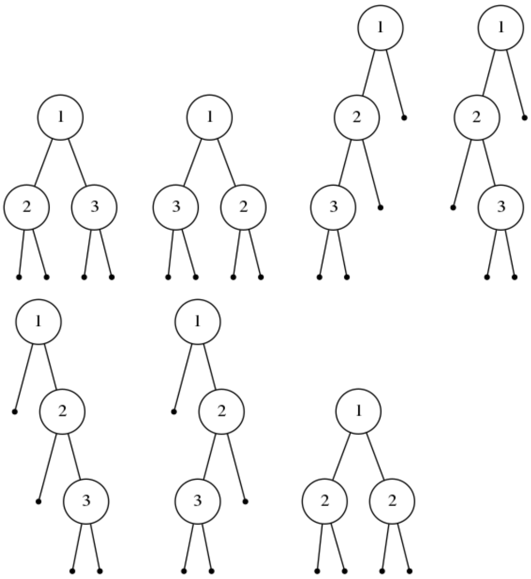


Figure 5.3: All 7 strict monotonic binary trees of size 4. There can be label repetitions, but the labellings along each branch are strictly increasing.

Example 5.3.2. *In* [**BGN19**] *and Section 4.3 we studied the class $\mathcal{G}$ ofstrict monotonic Schröder trees in which all degrees are allowed except unary nodes. The model corresponds to $r = \mathbb{N}^*$ and $\phi(z) = \frac{z^2}{1-z}$. The first values of G_n are:*

$$0, 1, 1, 3, 13, 75, 541, 4683, 47293, 545835, 7087261, \ldots$$

The sequence corresponds to Ordered Bell numbers also known as (Ordered set partitions). More information can be found in Section 3.2 and EIS A000670.

Example 5.3.3. *In* [**BGGW20**]*, the authors studied the class $\mathcal{T}$ of strict monotonic binary trees where $r = \mathbb{N}^*$ and $\phi(z) = z^2$. The first values of T_n are*

$$0, 1, 1, 2, 7, 34, 214, 1652, 15121, 160110, 1925442, \ldots$$

	r	$\phi(z)$	Asymptotics	References
S. M. Binary Trees	$\mathbb{N}^*$	z^2	$\alpha(n-1)!\, n^{-\ln 2}\, (\frac{1}{\ln 2})^n$	[BGGW20]
S. M. Binary-Ternary Trees	$\mathbb{N}^*$	z^2+z^3	$\kappa(n-1)!\, (\frac{1}{\ln 2})^n$	
S. M. Schröder trees	$\mathbb{N}^*$	$\frac{z^2}{1-z}$	$\frac{1}{2}\,(n-1)!\, (\frac{1}{\ln 2})^n$	[BGN19]

Table 5.3: An example of the change in behaviour of the asymptotics for different classes of our model. Here “S. M.” stands for “Strictly Monotonic”. The asymptotic behaviours all come from Theorem 5.2.6 and its specialisation to this case Corollary 5.2.10

The sequence is referenced under EIS A171792. The 7 *trees of size* 4 *can are drawn in Figure 5.3.*

The asymptotic regimes of both *strict monotonic Schröder trees* and *strict monotonic binary trees* are close since they only differ by a polynomial term namely $n^{-\ln 2}$. This led us to investigate when does the shift of behaviour appears. It turns out that adding ternary nodes (i.e $\phi(z) = z^2 + z^3$) suffices to have the asymptotic of $\phi(z) = \frac{z^2}{1-z}$. The results are summarised in Table 5.3.

Example 5.3.4. *Consider the class* $\mathcal{T}$ *with* $\phi(z) = z^2 + z^3$*. The first few values of* T_n *are:*

$$0, 1, 1, 3, 12, 68, 482, 4122, 41253, 472795, \dots$$

Then we have by Theorem 5.2.6

$$T_n \sim \kappa\,(n-1)!\,\left(\frac{2}{\ln 2}\right)^n.$$

with $\kappa \approx 0.41$ *as seen in simulations.*

Increasing binary trees with d repetitions

Example 5.3.5. *As a final example, we give the asymptotics growth of binary increasing trees with* d *label repetitions.*

Let $\mathcal{B}^d$ *be the class of increasing binary trees with* d *label repetitions at each iteration step. So that at each iteration step exactly* d *leaves are chosen to expand (we start with a single root that has* d *leaves). We can specify* $\mathcal{B}^d$ *from Equation (5.1) by taking* $\phi(z) = z^2$ *and* $r = \{d\}$*. For all* $d \geq 1$*, Theorem 5.2.6 is applicable. Some results are given using Theorem 5.2.6 in Table 5.4 and some simulations of the asymptotic behaviour in Figure 5.5.*

For instance the first values of B_n^2 *are:*

$$0, 0, 1, 0, 1, 0, 6, 0, 90, 0, 2520, 0, 113400, 0, 7484400, \dots$$

The 6 *trees of size* 6 *are depicted in Figure 5.4.*

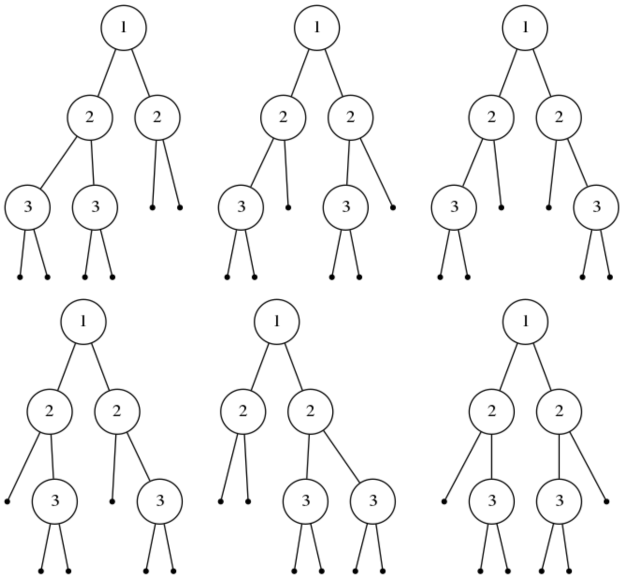


Figure 5.4: All 6 tree of increasing binary trees with 2 repetitions B_6^2.

d	*Asymptotics*	*References*
1	$(n-1)!$	*EIS A000142*
2	$c_2\, n!\, \left(2^{1/2}\right)^{-n} n^{-2}$	*EIS A000680*
3	$c_3\, n!\, \left(3!^{1/3}\right)^{-n} n^{-3}$	*EIS A014606*
4	$c_4\, n!\, \left(4!^{1/4}\right)^{-n} n^{-4}$	*EIS A014608*

Table 5.4: Asymptotic behaviour for B_n^d for $d \in \{1,2,3,4\}$ when $n \equiv 0 \bmod d$. The sequences in OEIS appear shifted (without periodicities).

$d=1$	1	1	2	6	24	120	720	5040
$d=2$	1	1	6	90	2520	113400	7484400	681080400
$d=3$	1	1	20	1680	369600	168168000	137225088000	182509367040000
$d=4$	1	1	70	34650	63063000	305540235000	3246670537110000	66475579247327250000

Table 5.5: First non zero values of increasing binary trees B_n^d with d repetitions and $d \in \{1,2,3,4\}$.

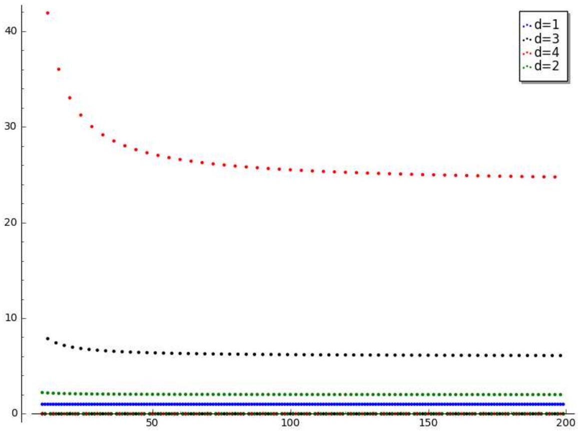


Figure 5.5: Simulation for $n \in \{10,200\}$ of the ratio of B_n^d and its main asymptotic behaviour with $d \in \{1,2,3,4\}$. Some coefficients are 0 because of the periodicities.

5.3.3 Varieties of connected monotonic Schröder trees

The definition of this labelling has been given in Section 3.5.6 and Definition 3.5.26. We recall it in the following

" A ***connected monotonic Schröder tree*** is a *labelled Schröder tree* such that the labellings are weakly increasing along branches and when a label i appears for the first time in the tree on node v (v is also the closest node to the root labelled i), then all other occurences of i appear in subtree of v. "

In fact generating a connected class of monotonic trees can be done using the *evolution process* in Definition 5.3.1. By taking $r = \{1\}$ and choosing the appropriate class of *tree shapes* in Equation (5.1) as described in Definition 5.3.1. So that, instead of colours the coloured degree function represents a class of rooted plane unlabelled trees counted by its number of leaves. Then we generate a variety of connected monotonic Schröder trees.

Example 5.3.6. *The class of* ***connected monotonic binary trees****. Let B be the class of plane binary trees with size equal to the number of leaves, given by*

$$\mathcal{B} = \mathcal{Z} + \mathcal{B}^2,$$

which translates to

$$B(z) = z + B(z)^2.$$

These numbers are counted by shifted Catalan numbers. By solving the above equation we find that $B(z) = \frac{1-\sqrt{1-4z}}{2}$. Now if we let $\mathcal{CB}$ to be the class of connected monotonic binary trees. Then by using the evolution process in Equation (5.1) parameterised with $\phi(z) = B(z) - z$ and $r = \{1\}$ we get:

$$CB(z) = z + CB'(z) \cdot (B(z) - z).$$

Then the first few values of CB_n, i.e the number of connected monotonic binary trees with $(n-1)$ internal nodes or with n leaves, are

$$0, 1, 1, 4, 21, 132, 958, 7872, 72273, 733772, 8167986, \ldots$$

We have depicted all 21 trees of size 4 in Figure 5.6. By Theorem 5.2.6, we have that

$$CB_n \sim \kappa_\phi \, n! \, n.$$

Example 5.3.7. *The class of* ***connected monotonic Schröder trees****. Let S be the class of Schröder trees (all arities except unary are allowed) which has the following specification,*

$$\mathcal{S} = \mathcal{Z} + \mathrm{Seq}_{\geq 2} \, \mathcal{S}.$$

By solving the above equation, we have $S(z) = \frac{1}{4}(1 + z - \sqrt{1 - 6z + z^2})$. The first terms of $S(z)$ are $z + z^2 + 3z^3 + 11z^4 + 45z^5 + 197z^6 + \ldots$. Hence. Let $\mathcal{CS}$ be the class of connected Schröder trees. The class $\mathcal{CS}$ is specified with parameters $r = \{1\}$ and $\phi(z) = S(z) - z$, then first values of CS_n, i.e., the number of connected monotonic Schröder trees with n leaves, are

$$0, 1, 1, 5, 32, 240, 2036, 19196, 199020, 2251764, 27630972, \ldots$$

We have by Theorem 5.2.6,

$$CS_n \sim \kappa_\phi \, n! \, n^2.$$

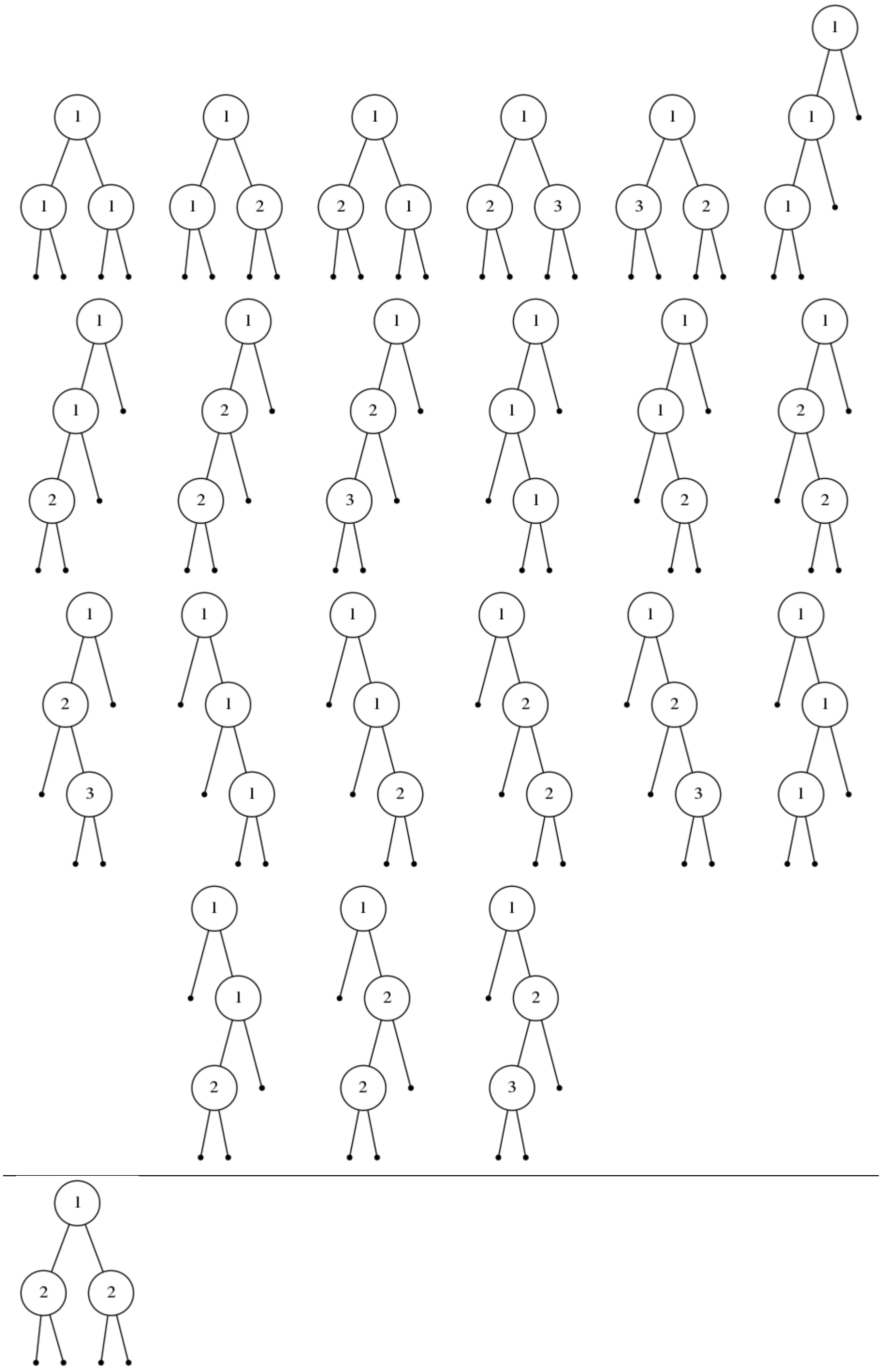


Figure 5.6: (Above) All 21 connected monotonic binary trees of size 4. (Below) This tree is the only tree of size 4 that belongs to the set of monotonic binary trees but not to the set of connected monotonic binary trees.

Let IB_n be the number of *increasing binary trees* with $n-1$ nodes. We have that $IB_n = (n-1)!$ and therefore

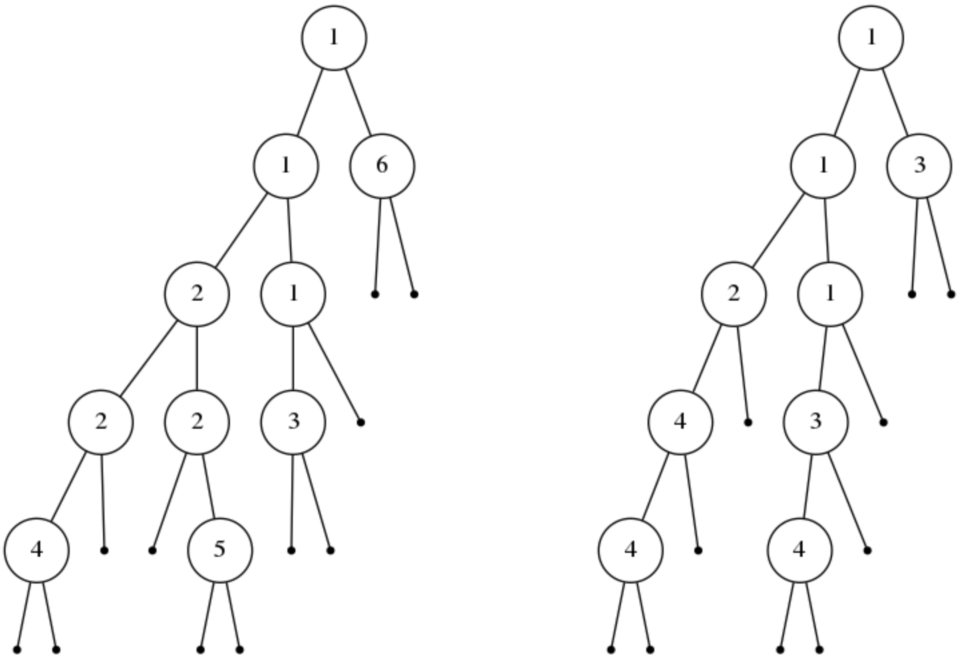


Figure 5.7: (left) A connected monotonic tree and (right) a monotonic binary tree. A monotonic binary tree need not to have the condition of connectedness. For instance, we see that the labels 3 and 4 appear in separate branches.

Proposition 5.3.8. *If we let, IB_n, be the number of increasing binary trees. Then,*

$$\frac{CS_n}{IB_n} \sim Cn^2,$$

for some positive constant C.

We also see the same asymptotic growth happens, between *increasing Schröder trees* and *connected monotonic Schröder trees* see Table 5.6 for a summary.

	r	$\phi(z)$	Asymptotics	References
Increasing Binary Trees	$\{1\}$	z^2	$(n-1)!$	[**FS09**],Theorem 5.2.6
C. M. Binary Trees	$\{1\}$	$(B(z)-z)$	$\alpha\, n!\, n$	Theorem 5.2.6
Increasing Schröder	$\{1\}$	$\frac{z^2}{1-z}$	$\frac{1}{2}\, n!$	[**BGN19**],Theorem 5.2.6
C. M. Schröder trees	$\{1\}$	$(S(z)-z)$	$\beta\, n!\, n^2$	Theorem 5.2.6

Table 5.6: Connected monotonic and increasing labelling for binary and Schröder trees. ‘C. M.” stands for “Connected Monotonic”

5.3.4 Varieties of monotonic Schröder trees

This form of labellings has been defined in Definition 3.5.23 and Section 3.5.6. We recall the definition

" A monotonic Schröder tree is a *labelled Schröder tree*, such such that the root node has label 1 and along each branch the labellings are weakly increasing and if m is the largest integer of the tree then all labels from 1 to m appear."

Let $\mathcal{T}$ be some class of unlabelled rooted plane trees counted by its number of leaves. We will denoted by $\mathcal{MT}$ the corresponding family of monotonic trees, i.e trees in $\mathcal{T}$ that have been labelled according to the rules for monotonic trees as defined in Definition 5.3.1. The idea here is to parameterise Equation (5.1) $\phi(z)$ corresponding to the variety of tree shapes that we want to label in a weakly increasing labelling along branches. We saw in the previous section that when $r = \{1\}$ we called these varieties connected monotonic Schröder trees. In the examples that we present in this section we always take $r = \mathbb{N}^*$ so that we can have any number of repetitions but this can be restricted be putting restrictions on the set r.

Example 5.3.9. *Consider the class $\mathcal{BT}$ of rooted plane binary-ternary unlabelled trees (whose size is their number of leaves). The specification of this class is*

$$\mathcal{BT} = \mathcal{Z} + \underset{\{2,3\}}{\mathrm{Seq}}\ \mathcal{BT}\,,$$

where the first terms are $BT(z) = z + z^2 + 3z^3 + 10z^4 + 38z^5 + 154z^6 + \ldots$. Then the first few values of MBT_n, i.e., the number of monotonic binary-ternary trees with n leaves, are

$$0, 1, 1, 5, 32, 252, 2340, 25048, 303862, 4121730, \ldots$$

Then $MBT(z)$ is defined with $r = \mathbb{N}^*$ and $\phi(z) = BT(z) - z$. By applying Theorem 5.2.6,

$$MBT_n \sim \kappa\ (n-1)! \left(\frac{1}{\ln 2}\right)^n n^{2\ln 2},$$

with $\kappa \approx 0.17$ as seen in simulations.

Example 5.3.10. *Let S be the class of Schröder trees (all arities except unary are allowed) which has the following specification,*

$$\mathcal{S} = \mathcal{Z} + \mathrm{Seq}_{\geq 2}\,\mathcal{S}.$$

By solving the above equation, we have $S(z) = \frac{1}{4}(1 + z - \sqrt{1 - 6z + z^2})$. The first terms of $S(z)$ are

$$S(z) = z + z^2 + 3z^3 + 11z^4 + 45z^5 + 197z^6 + \ldots$$

Hence, $MS(z)$ is defined with the paramters $r = \mathbb{N}^*$ and $\phi(z) = S(z) - z$, then the first values of MS_n, i.e., the number of monotonic Schröder trees with n leaves, are

$$0, 1, 1, 5, 33, 265, 2497, 27017, 330409, 4510065, \ldots$$

By Theorem 5.2.6 we have,

$$MS_n \sim \kappa\ (n-1)! \left(\frac{1}{\ln 2}\right)^n n^{2\ln 2},$$

with $\kappa \approx 0.19$.

	r	$\phi(z)$	*Asymptotics*	*References*
Increasing Schröder trees	$\{1\}$	$\frac{z^2}{1-z}$	$\frac{1}{2}\,n!$	[**BGN19**], *Theorem 5.2.6*
C. M. Schröder trees	$\{1\}$	$(S(z)-z)$	$\alpha\,n!\,n^2$	[**BGN19**],*Theorem 5.2.6*
Strictly monotonic Schröder	$\mathbb{N}^*$	$\frac{z^2}{1-z}$	$\frac{1}{2}(n-1)!\,(\frac{1}{\ln 2})^n$	[**BGN19**],*Theorem 5.2.6*
Monotonic Schröder	$\mathbb{N}^*$	$(S(z)-z)$	$\kappa(n-1)!\,(\frac{1}{\ln 2})^n\,n^{2\ln 2}$	*Theorem 5.2.6*

Table 5.7: Comparison of the asymptotic behaviour of families of labelled Schröder trees.

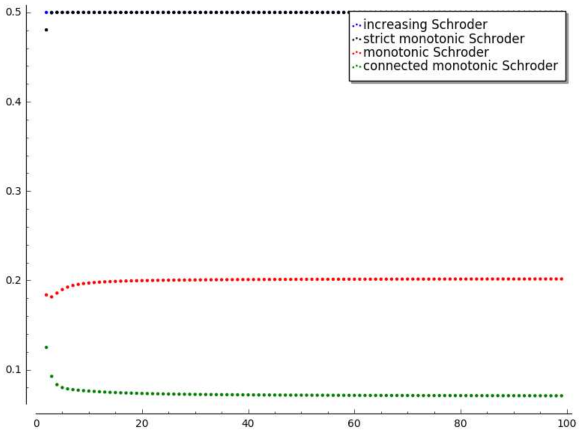


Figure 5.8: Simulation for $n \in \{1, 100\}$ of the ratio of Schröder trees with different labellings with their main asymptotic behaviour. The models of increasing Schröder trees and strict monotonic Schröder trees have the same constant and therefore their plots coincide.

5.3.5 Weakly increasing plane d-ary trees (monotonic d-ary trees)

It is a fact that our specification, by construction, enumerates families of trees by number of leaves. However there exists a special case, that of *monotonic d-ary Schröder trees* ($\mathcal{MT}$ where $\mathcal{T}$ is a variety of rooted plane m-ary trees), where our specification also allows for enumeration by number of internal nodes. In this specific case then, we are also able to enumerate by number of internal nodes since any d-ary tree with k leaves has $(k-1)/(d-1)$ internal nodes.

As an example, we consider the case of monotonic binary trees (an example is depicted in Figure 5.7). In this case we obtain the following.

Example 5.3.11. *Let $\mathcal{B}$ be the class of plane binary trees with size equal to the number of leaves, as defined in Section 5.3.3. Then $\mathcal{MB}$ is the class of* ***monotonic binary trees*** *(binary trees with weakly increasing labellings along branches). It is specified with $\phi(z) = B(z) - z$ and $r = \mathbb{N}^*$.*

	r	$\phi(z)$	Asymptotics	References
Labelled binary			$c_1\,(n-1)!\,4^n\,n^{-\frac{3}{2}}$	[FS09]
Weakly labelled			$c_2\,(n-1)!\,\left(\frac{4}{\ln 2}\right)^n\,n^{-\frac{3}{2}}$	
Increasing	$\{1\}$	z^2	$(n-1)!$	[FS09], Theorem 5.2.6
Connected monotonic	$\{1\}$	$(B(z)-z)$	$c_3\,n!\,n$	Theorem 5.2.6
Strict monotonic	$\mathbb{N}^*$	z^2	$c_4\,(n-1)!\,(\frac{1}{\ln 2})^n\,n^{-\ln 2}$	[BGGW20],Theorem 5.2.6
Monotonic (Weakly increasing)	$\mathbb{N}^*$	$(B(z)-z)$	$c_5\,(n-1)!\,(\frac{1}{\ln 2})^n\,n^{\ln 2}$	Theorem 5.2.6

Table 5.8: Comparison of the asymptotic behaviour of labelled binary trees under different labelling models.

The first few values of MB_n, i.e the number of monotonic binary trees with $(n-1)$ internal nodes and n leaves, are

$$0, 1, 1, 4, 22, 152, 1264, 12304, 137332, 1729584, 24265584, \ldots$$

By Theorem 5.2.6, we have that,

$$MB_n \sim \kappa\,(n-1)!\,\left(\frac{1}{\ln 2}\right)^n\,n^{\ln 2},$$

with $\kappa \approx 0.34$ as seen in simulations. We have depicted all 22 trees of size 4 in Figure 5.6.

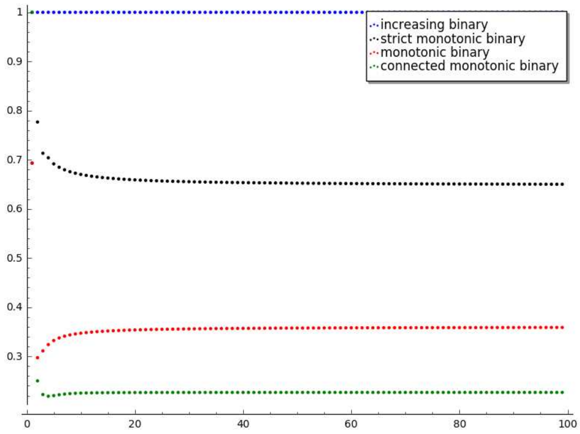


Figure 5.9: Simulation for $n \in \{1, 100\}$ of the ratio of binary trees with different labellings and their main asymptotic behaviour.

Finally, Table 5.7 and Table 5.8 summarises different results on the asymptotic behaviour of Schröder and binary trees with different forms of labellings. Figure 5.9 and Figure 5.8 shows the graph of the number of trees divided by the asymptotic behaviours obtained with Theorem 5.2.6. See Definition 3.5.5 for the weak labelling of a tree.

The *evolution process* we have presented, enumerates any class of plane weakly increasing d-ary trees (d-ary monotonic tree). However, the asymptotic enumeration in the theorems presented in Section 5.2 always require the existence of binary nodes.

Open question (Asymptotic enumeration). It is an open question as to give the asymptotic behaviour of plane weakly increasing d-ary trees. For $d = 2$, the problem falls under Theorem 5.2.6 and its specialisation Corollary 5.2.10. But for $d > 2$ the problem remains open. The parametrisation of Equation (5.1) is to take $r = \mathbb{N}^*$ and $\phi(z) = T_d(z)$ the class of rooted plane d-ary trees counted by its number of leaves. We have, for $d > 2$,

n	0	1	2	3	4	5	6	7	8	9	10	11
$d=2$	0	1	1	4	22	152	1264	12304	137332	1729584	24265584	375316704
$d=3$	0	1	0	1	0	6	0	54	0	638	0	9336
$d=4$	0	1	0	0	1	0	0	8	0	0	100	0

Table 5.9: First values of weakly increasing plane d-ary trees.

$$T_d(z) = z + (T_d(z))^d,$$

and then, the class $\mathcal{B}_d$ of plane weakly increasing d-ary trees is defined by:

$$B_d(z) = z + B_d(T_d(z) - z) - B_d(z).$$

We remind that when $r = \mathbb{N}^*$, the specification reduces to a substitution. The first values of $B_d(z)$ for $d = 2, 3, 4$ are presented in Table 5.9.

5.3.6 Applications of trees with unary nodes

In this section we give two examples of application of Theorem 5.2.7.

Example 5.3.12. *The class of $\mathcal{G}$ of general monotonic Schröder trees presented in Section 4.4 has $r = \mathbb{N}^*$ and $\phi(z) = \frac{z}{1-z}$. By Theorem 5.2.7, we have,*

$$\begin{aligned} G_n &\underset{n\to\infty}{\sim} \alpha\ (n-1)! \prod_{k=1}^{n-1} \left(\sum_{i=1}^{\infty} \binom{n-k-1}{i-1} \right) \\ &\underset{n\to\infty}{\sim} \alpha\ (n-1)! \prod_{k=1}^{n-1} \left(2^{n-k-1}\right) \\ &\underset{n\to\infty}{\sim} \alpha\ (n-1)!\, 2^{\sum_{k=1}^{n-1}(n-k-1)} \\ &\underset{n\to\infty}{\sim} \alpha\ (n-1)!\, 2^{(n-1)(n-2)/2}. \end{aligned}$$

This corresponds to the result found in Theorem 4.4.2.

Example 5.3.13. *Let us consider now the class $\mathcal{F}$ of trees, such that $r = \{1, 2\}$ and $\phi(z) = \frac{z}{1-z}$. The first values of F_n are:*

$$0, 1, 1, 5, 51, 883, 23285, 870911, 43913281, 2873499383, \ldots$$

By Theorem 5.2.7, we get,

$$\begin{aligned} F_n &\underset{n\to\infty}{\sim} \alpha\ (n-1)! \prod_{k=1}^{n-1} ((n-k)) \\ &\underset{n\to\infty}{\sim} \alpha\ (n-1)!^2. \end{aligned}$$

5.4 Combinatorial model

It is possible to write a general recurrence for these models using sets that are in bijection with integer partitions or integer compositions. We start by defining the following:

Definition 5.4.1. *[The set $A_{n,k,r,\phi}$.] We denote by $A_{n,r,k,\phi}$ the set of ordered multisets with elements in $\mathbb{N}$ such that for each ordered multiset $a = [a_1, \ldots, a_l] \in A_{n,k,r,\phi}$:*

- *We have that $a_1 + \cdots + a_l = k$.*
- *The elements $a_1, \ldots, a_l$ are ordered decreasingly.*
- *$|a| \in r$ and $|a| \leq (n-k)$.*
- *$\forall i, 1 \leq i \leq l, [z^{a_i+1}]\phi(z) > 0$.*

where $|a|$ represents the size of the list a.

The set In $A_{n,k,r,\phi}$ represents the different possiblities of making a subset of leaves grows to reach the final size n at an iteration step. In Figure 5.10, the set $A_{10,6,z^2/(1-z),\mathbb{N}^*}$, gives the different possiblities of a tree of size 4 to expand into one of size 10. The elements of the set A, give the configurations with different possible arities. The list $[3,2,1]$, says that we can expand by making 3 leaves evolve, one into a binary node, one into a ternary nodes and one into a quaternary node (there is a shift by one in the degrees). We will see in the recurrence here after that each configuration has then a certain weight that depends on its elements and the number of leaves in the tree, since it is possible to permute elements and get other trees.

$[2,2,1,1]$	$[3,1,1,1]$	$[2,2,2]$
$[3,2,1]$	$[4,1,1]$	$[3,3]$
$[4,2]$	$[5,1]$	$[6]$

Figure 5.10: An example of $A_{10,6,\phi,r}$ with $\phi(z) = \frac{z^2}{1-z}$ and $r = \mathbb{N}^*$.

Definition 5.4.2. *Let a be an ordered list of integers. We define the* maximum function *max to be the function that maps a to its greatest element. We define the* occurrences function *to be the one such that $occ(a) = [u_0, \ldots, u_{max(a)}]$ where u_i is the number of elements in a equal to i. For example, when $a = [4,3,1,1]$, $occ(a) = [0,2,0,1,1]$.*

Remark 5.4.3. *For simplicity we will write B_n instead of $B_n^{m,\phi}$ to make the notations lighter. But it is clear from the context that the two parameters always exist.*

Theorem 5.4.4. *For any weighted degree function $\phi(z)$ and $r \subset \mathbb{N}^*$. Let $m = min(r)$, The number of trees of size n generated by Equation (5.1) can be obtained using the following recurrence:*

$$B_n = \begin{cases} 0 & \textit{if } n < m \\ 1 & \textit{if } n = m \\ \sum_{k=1}^{n-1} \left(\sum_{a \in A_{n,k,r,\phi}} \frac{(n-k)!}{(n-k-|a|)!} \prod_{i=0,u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!} \right) B_{n-k} & \textit{if } n > m \end{cases} \tag{5.2}$$

Proof. We argue combinatorially. Fix some positive integer $1 \leq k \leq n-1$ and consider the number of trees of size n that can be constructed from a tree T of size $n-k$ by attaching to each of its leaves a number (possibly zero) of new vertices. Each element $a \in A_{n,k,\phi}$ then represents a way of building a tree of size n out of T by replaced some of its leaves with i-ary vertices. This procedure may be described as a linear ordering of the leaves of T such that for $1 \leq i \leq |a|$, the $i-th$ leaf of T is replaced by an $(a_i + 1)$-ary node, while for $i > |a|$, the i-th leaf in the ordering is left untouched. We now consider the multiplicative factors arising from the combinatorics of this procedure. To start with, we can naively suppose that after the aforementioned procedure, all leaves of T will have different new arities and so we can freely impose any order on them; this can be done in $(n-k)!$ ways. We will now proceed to refine this naive approach. First of all we have to account for the fact that some leaves of T do not get replaced and so may be freely permuted within a given order without affecting the resulting tree; this yields a factor of $\frac{1}{(n-k-|a|)!}$, since the number of leaves that actually change is $|a|$. In the case where $\phi_1 > 1$, some 0 integers will appear in a. But that is not a problem, since the size of a is constrained to be in r or smaller than $(n-k)$ which is the total number of leaves in B_{n-k}.

In the same vein, we have a factor

$$\prod_{i=0,u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!}$$

which accounts for the fact that for each arity i, the leaves that get replaced with i-ary vertices may be freely permuted among themselves in any given ordering, without affecting the outcome. Again, the factors $\frac{\phi_{i+1}^{u_i}}{u_i!}$ in last expression account for the fact that for each arity i we have to choose one of the ϕ_{i+1} colours for each of the u_i leaves that will be replaced by i-ary vertices. □

When unary nodes are not allowed (i.e $\phi_1 = 0$), we can write the sum in terms of restricted integer compositions.

Definition 5.4.5 (The set $C_{n,k,r,\phi}$.). *We denote by $C_{n,k,r,\phi}$ the set of restricted integer compositions defined as the set of ordered multisets with elements in $\mathbb{N}^*$ such that for each ordered multiset $a = [a_1, \ldots, a_l]$ and let j be the multiplicity of 1 in a (i.e j is the number of 1 appearing in a). Then $a \in C_{n,k,r,\phi}$ if:*

- *We have that $a_1 + \cdots + a_l = n$*
- $|a| = k$.

- $(|a| - j) \in r$.
- $\forall i, 1 \leq i \leq l$, *if* $a_i > 1, [z^{a_i}]\phi(z) > 0$.

where $|a|$ *represents the size of the list* a.

An example of this set is depicted in Figure 5.11.

$[6,1,1]$	$[5,2,1]$	$[5,1,2]$	$[4,3,1]$	$[4,1,3]$	$[2,2,4]$	$[2,3,3]$
$[1,6,1]$	$[2,5,1]$	$[1,5,2]$	$[3,4,1]$	$[1,4,3]$	$[2,4,2]$	$[3,2,3]$
$[1,1,6]$	$[2,1,5]$	$[1,2,5]$	$[3,1,4]$	$[1,3,4]$	$[4,2,2]$	$[3,3,2]$

Figure 5.11: An example of $C_{8,3,\phi,r}$ with $\phi(z) = \frac{z^2}{1-z}$ and $r = \mathbb{N}^*$.

$$B_n = \begin{cases} 0 & \text{if } n < m \\ 1 & \text{if } n = m \\ \sum\limits_{k=1}^{n-1} \left(\sum\limits_{a \in C_{n,n-k,r,\phi}} \prod\limits_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) B_{n-k} & \text{if } n > m \end{cases} \tag{5.3}$$

We know the number of ways to make a tree of size $n-k$ to evolve into a tree of size k, is the same number as the integer composition of n into $n-k$ parts. Each one of the $n-k$ leaves can be given a canonical ordering (from left to right for example) and then following the composition, into $n-k$ part, each leaf take a part. If the leaf is given a part which is 1 it does not change because it contributes to 1 size in the whole tree, if the leaf is given integer i with $i \geq 2$, then it will expand into an internal node with i new leaves. The integer composition has to be restricted according to the degree function $\phi(z)$. We saw that the leaves given integer 1 do not evolve, then the number of evolving leaves at an iteration step is $(|a| - j)$, and that number should be present in the set of allowed repetitions. Finally, for leaves evolving into internal nodes of same degree, they could take different colors according to $\phi(z)$ and thus the product arises from this situation.

We will mainly use Equation (5.3) in the proofs but Equation (5.2) has the advantage of working for any $\phi(z)$ and any r.

5.5 Asymptotic analysis for $r = \{d\}$

[Proof of Proposition 5.2.9].

This case is a generalisation that includes study of asymptotics enumeration of general Schröder trees as presented in Section 4.2 where $r = \{1\}$ and $\phi(z) = \frac{z^2}{1-z}$.

At each iteration step only one leaf evolves into an internal node with some new leaves. When $r = \{1\}$ the result to prove is

$$B_n^{\{1\},\phi} \sim \kappa_\phi \, n! \, {\phi_2}^n \, n^{\left(\frac{\phi_3}{\phi_2^2} - 1\right)}.$$

5.5.1 Asymptotic analysis for $r = \{1\}$

For simplification we will denote B_n instead of $B_n^{\{1\},\phi}$ in the proof. Our model can be specified via the symbolic method, as detailed in Section 2.2. Using Equation (5.2) we can directly obtain the following recurrence:

$$\begin{aligned} &B_1 = 1, \\ &B_n = \sum_{k=1}^{n-1} (n-k)\, \phi_{k+1}\, B_{n-k}. \end{aligned} \tag{5.4}$$

The main idea of the proof is to perform a Borel transform on the level of the coefficients of $B(z)$. Let $b(z) = \sum\limits_{n\geq 0} \frac{B_n}{(n-1)!} z^n$. We can thus get a new recurrence:

$$\begin{aligned} &b_1 = 1, \\ &b_n = \sum_{k=1}^{n-1} \frac{(n-k)!\, \phi_{k+1}}{(n-1)!}\, b_{n-k}. \end{aligned} \tag{5.5}$$

We remark that $\forall n \geq 2, b_n \geq b_{n-1}$. Indeed, if we develop the formula of b_n, we see that $b_n = \phi_2 b_{n-1} + \epsilon_n$ and all the terms in ϵ_n are positive. Let us define for all $n \geq 1, 1 \leq k \leq n-1$,

$$t_{n,k} = \frac{(n-k)!\, \phi_{k+1}}{(n-1)!}.$$

The idea is to leave ϕ_2 and ϕ_3 fixed and to let all other ϕ_n take their maximal value. We will see that whatever are the values of ϕ_n for $n > 3$ they only affect the constant in the first order asymptotic. We can show that,

Lemma 5.5.1. *There exists a constant c independent of n such that, $\forall n \geq 2$,*

$$\sum_{k=3}^{n-1} t_{n,k} \leq \frac{c}{(n-1)(n-2)}.$$

Proof. We know that there exist $n_0 \in \mathbb{N}^*$ and c_1, such that for all $n \geq n_0, \phi_n \leq c_1 \frac{n!}{n^5}$. Therefore, we can make a Stirling approximation on $t_{n,k}$, we get,

$$t_{n,k} \leq c_1 \frac{k(k+1)}{\binom{n-1}{k-1}(k+1)^5} \leq \frac{c_2}{\binom{n-1}{k-1}k^3}.$$

This works for large n and fixed k, but by positivity of ϕ_n, we can find a constant c_2 such that for any $3 \leq k \leq n-1$, $t_{n,k} \leq \frac{c_2}{\binom{n-1}{k-1}k^3}$. then,

$$\begin{aligned} t_{n,k} &\leq \sum_{k=3}^{n-1} \frac{c_2}{\binom{n-1}{k-1}k^3} = c_2 \sum_{k=3}^{n-1} \frac{1}{\binom{n-1}{k-1}k^3} \\ &\leq c_3 \frac{1}{n^2} + c_2 \sum_{k=4}^{n-1} \frac{1}{\binom{n-1}{k-1}k^3} = O\left(\frac{1}{n^2}\right). \end{aligned}$$

In the sum $\sum_{k=4}^{n-1} \frac{1}{\binom{n-1}{k-1}} = O\left(\frac{1}{n}\right)$, the factor k^3 comes to help getting the right bound. Therefore, there exists n_3 such that for all $n \geq n_3$, $\sum_{k=3}^{n-1} t_{n,k} \leq \frac{c}{(n-1)(n-2)}$. Since there are a finite number of steps until n_3, we can find a constant that works for all n, by taking the maximum from the smaller values.

□

We will define a new sequence which we call a *correcting sequence* defined as follows $a_1 = 1$ and for all $n \geq 2$

$$a_n = b_n - \phi_2 b_{n-1} - \frac{1}{(n-1)}\phi_3 b_{n-2}. \tag{5.6}$$

We can show that remainder sequence is

$$a_n \underset{n\to\infty}{=} O\left(\frac{b_n}{n^2}\right).$$

We have that by definition $a_n = \sum\limits_{k=1}^{n-3} t_{n,k} b_k$, and since b_n is increasing then

$$a_n \leq b_n \sum_{k=1}^{n-3} t_{n,k} = O\left(\frac{b_n}{n^2}\right),$$

by using Lemma 5.5.1. Now writing the generating function of b_n with the sequence of a_n we get

$$b_n = \phi_2 b_{n-1} + \frac{1}{(n-1)}\phi_3 b_{n-2} + a_n.$$

Which gives

$$\left(-\phi_3 z^2 - 1\right) b(z) + \left(-\phi_2 z^2 + z\right) b'(z) = z a'(z) - a(z).$$

The homogeneous part has the form

$$\left(\phi_3 z^2 - 1\right) y(z) + \left(-\phi_2 z^2 + z\right) y'(z) = 0.$$

Which solves to $y(z) = C g(z)$ with

$$g(z) = (-1)^{1+\frac{\phi_3}{\phi_2^2}} z\, \mathrm{e}^{-\frac{\phi_3 z}{\phi_2}} \left(1 - \phi_2 z\right)^{-1-\frac{\phi_3}{\phi_2^2}}.$$

By constant variation

$$-c'(z)\,(\phi_2 z - 1) z\, g(z) = z a'(z) - a(z).$$

We have $b(z) = c(z)\, g(z)$, therefore,

$$b(z) = g(z) \int\limits_0^z \frac{z a'(t) - a(t)}{(\phi_2 t - 1) t\, g(t)} dt.$$

We have,

$$g(z) \underset{z\to 1/\phi_2}{=} \frac{\mathrm{e}^{-\frac{\phi_3}{\phi_2{}^2}}}{\phi_2} \left(1 - \phi_2 z\right)^{-1-\frac{\phi_3}{\phi_2^2}} + O\left(\left(1 - \phi_2 z\right)^{-\frac{\phi_3}{\phi_2^2}}\right).$$

From which,

$$[z^n]g(z) = \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}\left(1+O\left(\frac{1}{n}\right)\right).$$

The coefficients of $za'(z) - a(z)$ are bounded above in absolute value by $K\,(1-\phi_2 z)^{-\frac{\phi_3}{\phi_2^2}}$. Finally,

$$\frac{za'(t) - a(t)}{(\phi_2 t - 1)t\, g(t)} \underset{z\to 1/\phi_2}{\sim} K_2,$$

For some constant K_2. From which,

$$h_n = [z^n]\int\limits_0^z \frac{za'(t) - a(t)}{(\phi_2 t - 1)t\, g(t)}dt = O\left(\frac{1}{n^{2+\epsilon}}\right).$$

Now,

$$b_n = \sum_{k=0}^{n} g_{n-k}h_k.$$

We see that,

$$\sum_{k=\lfloor n/2\rfloor}^{n} g_{n-k}h_k = O\left(\frac{g_n}{n^{1+\epsilon}}\right).$$

Since the smallest value for h_k is when $k = \frac{n}{2}$, and it is $h_{n/2} = O\left(\frac{1}{n^2}\right)$. For the second part of the sum,

$$\begin{aligned}
\sum_{k=0}^{\lceil n/2\rceil} g_{n-k}h_k &= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\phi_2}\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2^n \sum_{k=0}^{\lceil n/2\rceil} n^{\frac{\phi_3}{\phi_2^2}}\left(1+O\left(\frac{k}{n}\right)\right)h_k \\
&= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}\,{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}} \sum_{k=0}^{\lceil n/2\rceil}\left(1+O\left(\frac{k}{n}\right)\right)h_k \\
&= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}\,{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}\left(\left(\sum_{k=0}^{\infty} h_k - \sum_{k=\lfloor\frac{n}{2}\rfloor}^{\infty} h_k\right) + \sum_{k=0}^{\lceil n/2\rceil} O\left(\frac{k}{n}\right)\, h_k\right) \\
&= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}\,{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}\left(\left(\sum_{k=0}^{\infty} h_k - \sum_{k=\lfloor\frac{n}{2}\rfloor}^{\infty} h_k\right) + O\left(\frac{1}{n^{1+\epsilon}}\right)\right) \\
&= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}\,{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}\left(\left(\sum_{k=0}^{\infty} h_k - O\left(\frac{1}{n^{1+\epsilon}}\right)\right) + O\left(\frac{1}{n^{1+\epsilon}}\right)\right) \\
&= \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2}\,{\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}\left(\left(\sum_{k=0}^{\infty} h_k\right) + O\left(\frac{1}{n^{1+\epsilon}}\right)\right)
\end{aligned}$$

$$\underset{n\to\infty}{\sim} \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2} \left(\int_0^{1/\phi_2} \frac{za'(t)-a(t)}{(\phi_2 t-1)t\,g(t)}dt\right) {\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}}$$

Finally, we get,[4]

$$b_n = \frac{\mathrm{e}^{-\frac{\phi_3}{{\phi_2}^2}}}{\Gamma\left(1+\frac{\phi_3}{\phi_2^2}\right)\phi_2} \left(\int_0^{1/\phi_2} \frac{za'(t)-a(t)}{(\phi_2 t-1)t\,g(t)}dt\right) {\phi_2}^n\, n^{\frac{\phi_3}{\phi_2^2}} \left(1+O\left(\frac{1}{n}\right)\right).$$

To conclude, we know that,

$$B_n = n!\, b_n,$$

We see that B_n has then the right asymptotic behaviour, which concludes the Proof of Proposition 5.2.9.

The method that we have used in the end of this section with the remainder function is generic and it will be used to show the subsequent results.

5.5.2 Asymptotic analysis for $d \geq 2$

The proof of the case $r = \{1\}$ generalises with some modifications that we give in the following. We first restate the result to prove:

$$B_n^{\{d\},\phi} \sim \kappa\, n! \left(\frac{\phi_2}{d!^{1/d}}\right)^n n^{-d+d!^{1/d}\frac{\phi_3}{\phi_2^2}}.$$

Using Equation (5.2) we obtain the following recurrence:

$$B_n = \begin{cases} 0 & \text{if } n < d \\ 1 & \text{if } n = d \\ \displaystyle\sum_{k=1}^{n-1} \left(\left(\sum_{a\in C_{n,n-k,\{d\},\phi}} \prod_{i=2,u_i\neq 0}^{|occ(a)|} \phi_i^{u_i}\right)\right) B_{n-k}. & \text{if } n > d \end{cases} \tag{5.7}$$

In other words, when we have a tree of size $n-k$, the number of configurations to make a tree of size n are integer compositions such that the number of elements larger than one in the composition is equal to d and each element larger than in the composition is multiplied by the number of colours of the corresponding degree. For instance the first terms of B_n are

$$\begin{aligned} B_n = \phi_2^d \binom{n-d}{d} B_{n-d} + d\phi_2^{d-1}\phi_3 \binom{n-d-1}{d} B_{n-d-1} \\ + \phi_2^{d-2}\left(d\phi_2\phi_4 + \binom{d}{2}\phi_3^2\right)\binom{n-d-2}{d} B_{n-d-2} + \ldots \end{aligned}$$

Now we apply a modified *Borel transform* on the coefficients of B_n in order to normalise the term in front of B_{n-d}. This transform will also give us a new recurrence which its corresponding generating function is analytic. So that itis easier to extract an asymptotic behaviour from

[4] We are grateful to Professor Stephan Wagner for helping with the last calculations on a related problem.

it by using classical tools. We define for $n \geq 1$,

$$b_n = \frac{B_n}{(n-d)!\, d!^{-n/d}}.$$

Therefore the first terms of b_n are

$$b_n = \phi_2^d b_{n-d} + \frac{d\phi_2^{d-1}\phi_3}{(n-d)} b_{n-d-1} + \dots$$

We know that

$$\phi_n \underset{n\to\infty}{=} O\left(\frac{n!}{d!^{n/d}\, n^{4+d}}\right).$$

For a fixed k the maximal summand in

$$\sum_{a \in C_{n,n-k,\{d\},\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i},$$

is asymptotically $\phi_2^{d-1}\phi_{k+2-(d-1)}$ which corresponds to the configuration which have $(d-1)$ binary nodes and a node of size $k+2-(d-1)$ because ϕ_n grows factorially. See Lemma 5.6.5 for a similar discussion. We conclude that for a fixed $k > d+1$

$$\sum_{a \in C_{n,n-k,\{d\},\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} = O\left(\binom{n-1}{d-1}\phi_2^{d-1}\phi_{k+2-(d-1)}\right)$$

The factor $\binom{n-1}{d-1}$, comes from the fact the number of summands is bounded by integer compositions of n with size d. Let us define for all $n \geq 1, d+2 \leq k \leq n-d$,

$$t_{n,k} = \phi_2^{d-1}\phi_{k+2-(d-1)}\binom{n-1}{d-1}\frac{(n-d-k)!d!^{-(n-d-k)/d}}{(n-d)!d!^{-(n-d)/d}}.$$

$$\begin{aligned}
t_{n,k} &\leq \phi_2^{d-1}\frac{(k+2-(d-1))!}{d!^{(k+2-(d-1))/d}(k+2-(d-1))^{d+4}}\;\frac{(n-1)!(n-d-k)!d!^{-(n-d-k)/d}}{(n-d)!^2\,(d-1!)d!^{-(n-d)/d}}\\
&\leq d!^{-3/d}\phi_2^{d-1}\frac{(k+2-(d-1))!}{(k+2-(d-1))^{d+4}}\;\frac{d\,(n-1)!(n-d-k)!}{(n-d)!^2}\\
&\leq d!^{-3/d}\phi_2^{d-1}\, d\,\frac{(n-1)^{\underline{d-1}}}{k^{\underline{d-3}}(k+3-d)^{d+4}}\;\frac{1}{\binom{n-d}{k}}
\end{aligned}$$

By a crude estimate we find that $t_{n,k} = O\left(1/n^3\right)$ and therefore

$$\sum_{k=d+2}^{n-d} t_{n,k} = O\left(\frac{1}{n^2}\right).$$

The sequence of b_n is increasing in each one of its d regimes because of its recurrence $b_n = \phi_2^d b_{n-d} + \epsilon_n$, where ϵ_n contains only positive terms.

From the last expression we define for $n \geq 1$,

$$b_n = \phi_2^d \underline{b}_{n-d} + \frac{d!^{1/d} d\phi_2^{d-1} \phi_3}{(n-d)} \underline{b}_{n-d-1} + a_n$$

From which by coefficient translation we get

$$\left(-d!^{1/d}\, d{\phi_2}^{d-1}\phi_3 z^{d+1} - d\right) b\left(z\right) + \left(-{\phi_2}^d z^{d+1} + z\right) \partial\, b\left(z\right) = za'(z) - a(z).$$

We can solve the homogeneous part

$$\left(-d!^{1/d}\, d{\phi_2}^{d-1}\phi_3 z^{d+1} - d\right) y\left(z\right) + \left(-{\phi_2}^d z^{d+1} + z\right) \partial\, y\left(z\right) = 0.$$

Which solves to $y(z) = Cg(z)$ with

$$g(z) = h(z)\ \left(1 - \phi_2 z\right)^{-1-d!^{1/d} \frac{\phi_3}{\phi_2^2}}.$$

with $h(z)$ is analytic at $z = 1/\phi_2$. The result is obtained since the real dominant singularity is invariably $\zeta = \frac{1}{\phi_2}$ and is a regular one, there are d singularities of the same modulus. Then the contribution of the real singularity can be computed mechanically with

$$\delta_1 = \lim_{z \to \zeta} (z - \zeta) d_1(z) = 1 + \frac{d!^{1/d}\phi_3}{\phi_2^2}.$$

From the estimation of $t_{n,k}$ we find that $a_n = O(\sum_{i=0}^{d} b_{n-i}/n^2)$. Since each regime in b_n is increasing. Then following the same calculations as in the end of Section 5.5.

The sequence b_n can be periodic and this is reflected by the presence of multiple singularities on the radius of convergence. However all singularities give the same asymptotic behaviour and we have three possible cases. Either b_n is periodic with 0 coefficients at regular intervals and in this case the contribution of the singularities cancel each others. There can also be different asymptotic constants so that all regimes have the same asymptotic behaviour with different constants or finally the constants of the different regimes converge to the same constant and the first order asymptotic of b_n has one constant. But in all cases when $n \equiv 0$ mod d and n tends to ∞

$$b_n \underset{n \to \infty}{\sim} \alpha\, \phi_2^n\, n^{d!^{1/d} \frac{\phi_3}{\phi_2^2}},$$

for some positive constant α. Then the main asymptotic order of B_n is

$$B_n = b_n\, (n-d)!\, d!^{-n/d},$$

which is the desired result.

5.6 Asymptotic analysis for $r = \mathbb{N}^*$

We remind that here $\phi_1 = 0.$ and $r = \mathbb{N}^*$, the specification in Equation (5.1) reduces to a substitution,

$$B(z) = z + B(z + \phi(z)) - B(z)$$

Since combinatorially, any subset of leaves can be chosen to expand at each iteration step. Thus we can put a substitution and add z inside the substitution so that any subset of leaves can evolve. The problem with this addition is that we need to remove the case where not any leaf has expanded and therefore the factor $-B(z)$ appears to account for this fact. We can write a recurrence on the coefficient of B_n which is Equation (5.2) that we recall here. In fact we have two recurrences,

$$\begin{aligned} B_1 &= 1, \\ B_n &= \sum_{k=1}^{n-1} \left(\sum_{a \in A_{n,k,\phi}} \frac{(n-k)!}{(n-k-|a|+1)!} \prod_{i=2,u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!} \right) B_{n-k}, \end{aligned} \tag{5.8}$$

It will be useful for us to give a name to the inner sum of the recurrence. Let

$$t_{n,k} = \sum_{a \in C_{n,n-k,r,\phi}} \prod_{i=2,u_i \neq 0}^{|occ(a)|} \phi_i^{u_i}.$$

Lemma 5.6.1. *If for all $n \geq 1$ and if for all $i \geq 2$, if $\phi_i = 1$ then*

$$t_{n,k} = \binom{n-1}{k} = \binom{n-1}{n-k-1}.$$

Proof. In this case, we have all repetitions allowed and all arities allowed, so that the set $C_{n,n-k,r,\phi} = C_{n,n-k}$,where $C_{n,n-k}$ is the integer compositions of n into $n-k$ parts and $|C_{n,n-k}| = \binom{n-1}{n-k-1}$. See Section 3.3 for more on *integer compositions.* □

We can rewrite the recurrence of B_n by taking out some of the factors and rearranging the rest of the terms.

$$\begin{aligned} B_n =& \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \binom{n-k}{k} B_{n-k} \\ &+ \sum_{k=2}^{\lceil \frac{n}{2} \rceil} \phi_2^{k-2} \phi_3 (k-1) \binom{n-k}{k-1} B_{n-k} \\ &+ \sum_{k=3}^{n-1} \left(\left(\sum_{a \in C_{n,n-k,r,\phi}} \prod_{i=2,u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \right. \\ &\left. - \phi_2^k \binom{n-k}{k} - \phi_2^{k-2} \phi_3 (k-1) \binom{n-k}{k-1} \right) B_{n-k} \end{aligned} \tag{5.9}$$

Combinatorially, in the recurrence it is possible to separate the terms that involve only adding binary nodes, and the ones that involve adding a single ternary and binary nodes from the rest of the terms.

In the last recurrence when $1 \leq k \leq \lfloor \frac{n}{2} \rfloor$ the terms $\phi_2^k \binom{n-k}{k}$ exists. If $2 \leq k \leq \lceil \frac{n}{2} \rceil$ the term $\phi_2^{k-2} \phi_3 \binom{n-k}{k-1}$ exists. But these two sums can be extended to n since the additional terms are 0.

$$\begin{aligned}B_n =& \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \binom{n-k}{k} B_{n-k} \\ &+ \sum_{k=2}^{\lceil \frac{n}{2} \rceil} \phi_2^{k-2} \phi_3 (k-1) \binom{n-k}{k-1} B_{n-k} \\ &+ \sum_{k=3}^{n-1} \left(\left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \right) B_{n-k}.\end{aligned} \tag{5.10}$$

Remark 5.6.2. *In the recurrence, $\mathbf{C}_{n,n-k,r,\phi}$ is $C_{n,n-k,r,\phi}$ where the configurations with k integers larger than 1 or $(k-1)$ integers larger than 1 has been removed from the set*

Since we think that these two first terms are the only ones which play a role in the asymptotic first order we will look for an estimate for the rest of the terms. The following two Lemmas will serve as a basis to write the main term of the differential equation that satisfies the generating function of B_n.

Lemma 5.6.3. *$\forall n \geq 1, 1 \leq k \leq n$,*

$$\frac{1}{k!} \left(1 - \frac{k^2}{n} - \frac{k^4}{n^2} \right) \leq \frac{(n-k)!^2}{k!\; n!\; (n-2k)!} \leq \frac{1}{k!} \left(1 - \frac{k^2}{n} + \frac{k^4}{n^2} \right).$$

Proof. Let us start with the upper bound, we notice that for $k > \lfloor \frac{n}{2} \rfloor$, $\frac{(n-k)!^2}{k!n!(n-2k)!} = 0$ and $\frac{k^2}{n} \geq 1$ therefore $\frac{k^4}{n^2} = (\frac{k^2}{n})^2 \geq \frac{k^2}{n}$ and finally $\frac{1}{k!}(1 - \frac{k^2}{n} + \frac{k^4}{n^2}) \geq 0$. Now we need to show the upper bound for the rest, that is when $1 \leq k \leq \lfloor \frac{n}{2} \rfloor$

$$\begin{aligned}\frac{(n-k)!^2}{k!n!(n-2k)!} &= \frac{1}{k!} \frac{(n-k)(n-k-1)\dots(n-2k+1)}{n(n-1)\dots(n-k+1)} \\ &= \frac{1}{k!}(1-\frac{k}{n})(1-\frac{k}{n-1})\dots(1-\frac{k}{n-k+1}) \\ &\geq \frac{1}{k!}\left(1 - k\left(\frac{1}{n} + \frac{1}{n-1} + \dots + \frac{1}{n-k+1}\right)\right) \\ &= \frac{1}{k!}\left(1 - k\left(H_n - H_{n-k}\right)\right) \\ &\geq \frac{1}{k!}\left(1 - k\left(\ln n - \ln(n-k)\right)\right) \\ &= \frac{1}{k!}\left(1 - k \ln\left(\frac{1}{1-\frac{k}{n}}\right)\right)\end{aligned}$$

In the proof we have used the fact that $(H_n - \ln n)_n$ is monotonically decreasing see [TT71] for example. We only look at values for $x \leq \frac{1}{2}$ and in this range $\ln \frac{1}{1-x} \leq x + x^2$, then,

$$\frac{(n-k)!^2}{k!n!(n-2k)!} \geq \frac{1}{k!}\left(1 - \frac{k^2}{n} - \frac{k^4}{n^2}\right).$$

For the other side,

$$\begin{aligned}\frac{(n-k)!^2}{k!n!(n-2k)!} &= \frac{1}{k!}\frac{(n-k)(n-k-1)\dots(n-2k+1)}{n(n-1)\dots(n-k+1)}\\ &= \frac{1}{k!}(1-\frac{k}{n})(1-\frac{k}{n-1})\dots(1-\frac{k}{n-k+1})\\ &\leq \frac{1}{k!}\left(1-\frac{k}{n}\right)^k\\ &\leq \frac{1}{k!}\left(1-\frac{k^2}{n}+\frac{k^4}{n^2}\right)\end{aligned}$$

The last inequality results from the fact that $(1-x)^r \leq 1 - rx + \binom{r}{2}x^2$. □

Lemma 5.6.4. $\forall n \geq 1, 1 \leq k \leq n$

$$\frac{1}{k!}\left(\frac{k(k-1)}{n} - \frac{k^2(k-1)^2}{n^2}\right) \leq \frac{(n-k)!^2}{n!(n-2k+1)!(k-2)!} \leq \frac{1}{k!}\left(\frac{k(k-1)}{n}\right).$$

Proof.

$$\begin{aligned}\frac{(n-k)!^2}{n!(n-2k+1)!(k-2)!} &= \frac{1}{(k-2)!}\frac{(n-k)\dots(n-2k+2)}{n(n-1)\dots(n-k+1)}\\ &= \frac{1}{n(k-2)!}(1-\frac{k-1}{n-1})(1-\frac{k-1}{n-2})\dots(1-\frac{k-1}{n-k+1}).\end{aligned}$$

We want that $\forall n, 1 \leq k \leq n$,

$$\begin{aligned}&\frac{1}{n(k-2)!}(1-\frac{k-1}{n-1})(1-\frac{k-1}{n-2})\dots(1-\frac{k-1}{n-k+1})\\ &\qquad \geq \frac{1}{n(k-2)!}(1-\frac{k(k-1)}{n}).\end{aligned}$$

We see that if $k \in \{\lceil \frac{n}{2} \rceil + 1, \dots, n\}$ the left hand side is equal to 0 and the right hand side is negative so the result holds. Now it is enough to show that $\forall n, 1 \leq k \leq \lceil \frac{n}{2} \rceil$,

$$\frac{(1-\frac{k-1}{n-1})(1-\frac{k-1}{n-2})\dots(1-\frac{k-1}{n-k+1})}{(1-\frac{k(k-1)}{n})} \geq 1.$$

We have that,

$$\begin{aligned}\frac{(1-\frac{k-1}{n-1})(1-\frac{k-1}{n-2})\dots(1-\frac{k-1}{n-k+1})}{(1-\frac{k(k-1)}{n})} &\geq \frac{\left(1-\frac{k-1}{n-k+1}\right)^{k-1}}{(1-\frac{k(k-1)}{n})}\\ &\geq \frac{\left(1-\frac{(k-1)^2}{n-k+1}\right)}{(1-\frac{k(k-1)}{n})}\\ &= \frac{n}{n-k+1} \geq 1\end{aligned}$$

For the other side we have,

$$\frac{(n-k)!^2}{n!(n-2k+1)!(k-2)!} = \frac{1}{(k-2)!}\frac{(n-k)\dots(n-2k+2)}{n(n-1)\dots(n-k+1)}$$

$$\begin{aligned}
&= \frac{1}{n(k-2)!}(1-\frac{k-1}{n-1})(1-\frac{k-1}{n-2})\dots(1-\frac{k-1}{n-k+1}).\\
&\leq \frac{1}{n(k-2)!}(1-\frac{k-1}{n-1})^{k-1}.\\
&\leq \frac{1}{n(k-2)!}.\\
&= \frac{1}{k!}\frac{k(k-1)}{n}\\
&\leq \frac{1}{k!}\left(\frac{k(k-1)}{n}+\frac{k(k-1)}{n^2}\right)
\end{aligned}$$

where the last inequality follows from the fact that $\frac{k(k-1)}{n^2} \geq 0$ for all $n \geq 0$ and $1 \leq k \leq n$. □

The following calculations help getting a bound to estimate the rest of the terms in the recurrence. The idea is to find an upper bound for the rest of the terms and then show that it is bounded.

We want to get a good upper bound for the rest of the terms that are of the form

$$\sum_{a\in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2,u_i\neq 0}^{|occ(a)|} \phi_i^{u_i}.$$

The first Lemma gives an upper bound for the value of the greatest element of this sum.

Lemma 5.6.5. *For $n \geq 3$, and $3 \leq k \leq n-1$. If $\phi_n = O\left(\frac{n!}{n^5}\right)$. Let mx be the maximal summand of*

$$\sum_{a\in \boldsymbol{C}_{n,n-k,r,\phi}} \prod_{i=2,u_i\neq 0}^{|occ(a)|} \phi_i^{u_i}.$$

There exists a positive constant c independent of n, such that

$$mx \leq c\frac{k!}{k^4}.$$

Proof. For a fixed k, and as n grows, the optimal configuration (the one that gives the largest product) is the list containing 1 everywhere except for one element which has value $(k+1)$. Because of the factorial growth on ϕ_n.

For example if for $\phi_2 = 7$ and for all $i > 2, \phi_i = (i-2)!$. If $k = 3$, when $n = 4$, the only composition possible is $[4]$ which gives $2! = 2$. But when $n = 5$, the composition $[2, 3]$ gives 7 while $[4, 1]$ gives $2! = 2$. Then when $n = 6$, the optimal composition is $[2, 2, 2]$ which gives $7^3 = 343$ while $[4, 1, 1]$ gives 2 and $[3, 1, 2]$ gives 7. and then we see that the optimal one is always $[2, 2, 2, 1^{n-k-3}]$, where 1^{n-k-3} means that we add $n-k-3$ ones to the list. And the product always have the same value. This comes from the fact at the beginning there were not enough leaves for the optimal configuration to settle, but ones it is done the product of the optimal configuration stabilises.

Now, we want to show that, there exists n_0 and $1 \leq \eta \leq n-1$, such that for all $n \geq n_0$ and $\eta \leq k \leq n-1$, the optimal configuration is $[k+1, 1^{n-k-1}]$ and when $k < \eta$, the optimal configurations are stable which means that when n grows we only add more and more ones and that does not change the overall product.

This result follows from the conditions on $\phi(z)$. Since other configurations involve product of factorials smaller than $(k+1)$ but which still have the same number of factors if the factorials are flattened. These compositions have the form

$$p = p_1! \times p_2! \times \cdots \times p_i!,$$

where p_j are all ≥ 2, and is such that $(\sum_{j=1}^{i} p_j) = k + i$, and

$$p = O\left(\frac{k!}{k^4}\right),$$

and the only configuration that reaches $O\left(\frac{k!}{k^4}\right)$ is $[k+1, 1^{n-k-1}]$. Whereas for $k < \eta$, the largest product can have another form due to the first values of $\phi(z)$. Therefore, after a finite n_1 and $\eta \leq k \leq n-1$ the optimal composition is $[k+1, 1^{n-k-1}]$.

For $k < \eta$, as it has been said the optimal compositions stabilise when n grows. Since η is finite, there exists n_2 such that for all $n \geq n_2$, and $k < \eta$ the optimal configuration are stable because they have reached the maximal product. In fact, these optimal compositions stabilise as soon as $n - k$ has a number of leaf large enough for the optimal composition to settle.

Finally, we take $n_0 = max(\{n_1, n_2\})$. Then, starting from $n \geq n_0$, and $k < \eta$, the optimal composition is stable (does not depend on n), we only add 1 for each growing n but the overall product does not change. When $k \geq \eta$ As n grows k can take larger values, but for all these values we know the optimal composition that is $[k+1, 1^{n-k-1}]$.

In order to determine the value of the constant c we have for a finite number of values $n \leq n_0$ and $3 \leq k \leq n-1$, where we divide each one of them by $\frac{k!}{k^4}$ and take the maximum value as a general constant that will work for all n. An example is depicted in Section 7.5.1. □

Let us now define the ***Borel transform*** of B_n, for all $n \geq 1$,

$$b_n = \frac{B_n}{n!}.$$

By replacing in the above equations we get,

$$\begin{aligned} b_n = & \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \binom{n-k}{k} \frac{(n-k)!}{n!} b_{n-k} \\ & + \sum_{k=2}^{\lceil \frac{n}{2} \rceil} \phi_2^{k-2} \phi_3 (k-1) \binom{n-k}{k-1} \frac{(n-k)!}{n!} b_{n-k} \\ & + \sum_{k=3}^{n-1} \left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \frac{(n-k)!}{n!} b_{n-k} \end{aligned} \tag{5.11}$$

$$
\begin{aligned}
b_n \leq & \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \binom{n-k}{k} \frac{(n-k)!}{n!} b_{n-k} \\
& + \sum_{k=2}^{\lceil \frac{n}{2} \rceil} \phi_2^{k-2} \phi_3 (k-1) \binom{n-k}{k-1} \frac{(n-k)!}{n!} b_{n-k} \\
& + \sum_{k=3}^{n-1} \left(c \, \frac{k!}{k^4} \left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} 1 \right) \right) \frac{(n-k)!}{n!} b_{n-k}
\end{aligned}
$$

In this way it is possible to define $\overline{b}_n$ and $\underline{b}_n$,

The last sum arise by applying Lemma 5.6.5. We need now to estimate the size of the set $\mathbf{C}_{n,n-k,r,\phi}$.

Lemma 5.6.6. *For $n \geq 4$, $3 \leq k \leq n-1$,*

$$
\left(\sum_{a \in \boldsymbol{C}_{n,n-k,r,\phi}} 1 \right) \frac{(n-k)!}{n!} = O\left(\frac{k^4}{k!\, n^2} \right).
$$

Proof. We know that $|C_{n,n-k,r,\phi}| \leq |C_{n,k}| = \binom{n-1}{k}$. Therefore,

$$
|\mathbf{C}_{n,n-k,r,\phi}| \leq \binom{n-1}{k} - \binom{n-k}{k} - (k-1)\binom{n-k}{k-1}.
$$

We already have lower bounds for $\binom{n-k}{k}\frac{(n-k)!}{n!}$ and $(k-1)\binom{n-k}{k-1}\frac{(n-k)!}{n!}$ with Lemma 5.6.3 and Lemma 5.6.4. We have for the first term,

$$
\binom{n-1}{k} \frac{(n-k)!}{n!} = \frac{1}{k!} - \frac{1}{(k-1)!\, n}.
$$

Therefore,

$$
\begin{aligned}
& \left(\binom{n-1}{k} - \binom{n-k}{k} - (k-1)\binom{n-k}{k-1} \right) \frac{(n-k)!}{n!} \\
& \leq \frac{1}{k!} - \frac{1}{(k-1)!\, n} - \frac{1}{k!}\left(1 - \frac{k^2}{n} - \frac{k^4}{n^2}\right) - \frac{1}{k!}\left(\frac{k(k-1)}{n} - \frac{k^2(k-1)^2}{n^2} \right) \\
& = O\left(\frac{k^4}{k!\, n^2} \right).
\end{aligned}
$$

□

Therefore, from Lemma 5.6.6 and Lemma 5.6.5, for $n \geq 4$, and $3 \leq k \leq n-1$,

$$
\left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \frac{(n-k)!}{n!} = O\left(\frac{1}{n^2} \right). \tag{5.12}
$$

In order to conclude the equivalent result that we are looking to demonstrate we will define a new sequence that is a remainder one and analyse it. Let us write b_n with a remainder term,

$$b_n = \left(\sum_{k=1}^{n} \left(\frac{\phi_2^k}{k!} \left(1 + \left(\frac{\phi_3}{{\phi_3}^2} - 1 \right) \frac{k^2}{n} - \frac{\phi_3}{{\phi_2}^2} \frac{k}{n} \right) \right) b_{n-k} \right) + a_n. \tag{5.13}$$

From which by splitting the sum as before we find,

$$\begin{aligned} a_n =& \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \left(\binom{n-k}{k} \frac{(n-k)!}{n!} - \frac{1}{k!} \left(1 - \frac{k^2}{n} \right) \right) b_{n-k} \\ &+ \sum_{k=2}^{\lceil \frac{n}{2} \rceil} \phi_2^{k-2} \phi_3 \left((k-1) \binom{n-k}{k-1} \frac{(n-k)!}{n!} - \frac{1}{k!} \left(\frac{k(k-1)}{n} \right) \right) b_{n-k} \\ &- \sum_{k=\lfloor \frac{n}{2} \rfloor}^{n-1} \phi_2^k \left(\frac{1}{k!} \left(1 - \frac{k^2}{n} \right) \right) b_{n-k} \\ &- \sum_{k=\lceil \frac{n}{2} \rceil}^{n-1} \phi_2^{k-2} \phi_3 \frac{1}{k!} \left(\frac{k(k-1)}{n} \right) b_{n-k} \\ &+ \sum_{k=3}^{n-1} \left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \frac{(n-k)!}{n!} b_{n-k}. \end{aligned} \tag{5.14}$$

Lemma 5.6.7. *The remainder sequence is asymptotically*

$$a_n \underset{n \to \infty}{=} O\left(\frac{b_n}{n^2} \right).$$

Proof. Looking at Equation (5.14), the first two sums are bounded by Lemma 5.6.3 and Lemma 5.6.3. For instance the first sum, we have,

$$\phi_2^k \left(\binom{n-k}{k} \frac{(n-k)!}{n!} - \frac{1}{k!} \left(1 - \frac{k^2}{n} \right) \right) = O\left(\frac{\phi_2^k \, k^4}{k! \, n^2} \right),$$

and therefore,

$$\begin{aligned} \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \phi_2^k \left(\binom{n-k}{k} \frac{(n-k)!}{n!} - \frac{1}{k!} \left(1 - \frac{k^2}{n} \right) \right) b_{n-k} &\leq \frac{b_n}{n^2} \sum_{k=1}^{\lfloor \frac{n}{2} \rfloor} \frac{c \, \phi_2^k \, k^4}{k!} \\ &= O\left(\frac{b_n}{n^2} \right). \end{aligned}$$

In the proof c is some positive constant. The second sum in Equation (5.14) can be treated in the same way. as well as the last sum by Equation (5.12). The two sums in the middle starts at $n = \lfloor n/2 \rfloor$. For example,

$$\sum_{k=\lfloor \frac{n}{2} \rfloor}^{n-1} \phi_2^k \left(\frac{1}{k!} \left(1 - \frac{k^2}{n} \right) \right) = O\left(\frac{1}{(\lfloor n/2 \rfloor - 2)!} \right),$$

Therefore,

$$\begin{aligned}\sum_{k=\lfloor\frac{n}{2}\rfloor}^{n-1}\phi_2^k\left(\frac{1}{k!}\left(1-\frac{k^2}{n}\right)\right)b_{n-k} &\leq b_n\sum_{k=\lfloor\frac{n}{2}\rfloor}^{n-1}\frac{c_2}{(\lfloor n/2\rfloor-1)!}\\ &\leq b_n\frac{c_2}{(\lfloor n/2\rfloor-2)!}\\ &=O\left(\frac{b_n}{n^2}\right),\end{aligned}$$

where c_2 is some positive constant. The same can be done for the remaining sum. Therefore, the result holds. □

From Equation (5.13) a differential equation for $b(z)$ can be written in term of the function $a(z)$:

$$-\mathrm{e}^{\phi_2 z}z\left({\phi_2}^2-\phi_3-\frac{\phi_2}{z}\right)b\left(z\right)+\partial_z a\left(z\right)+\left(\mathrm{e}^{\phi_2 z}-2\right)\partial_z b\left(z\right),$$

which its homogeneous part has a generic solution $b(z)=C\,g(z)$ and $g(z)$ is:

$$g(z)=\exp\left(\int\limits_0^z\frac{\mathrm{e}^{\phi_2 t}t\left({\phi_2}^2-\phi_3-\frac{\phi_2}{t}\right)}{\left(\mathrm{e}^{\phi_2 t}-2\right)}\,dt\right).$$

The function $g(z)$ can be expanded around its singularity $\frac{\ln 2}{\phi_2}$ to give:

$$g(z)\underset{z\to\ln 2/\phi_2}{\sim}\exp\left(-\frac{\left({\phi_2}^2-\phi_3\right)\ln\left(2\right)\left(\ln\left(\ln\left(2\right)\right)-\ln\left(-z\phi_2+\ln\left(2\right)\right)\right)}{{\phi_2}^2}\right),$$

which gives asymptotically

$$g_n\underset{n\to\infty}{\sim}\frac{1}{\Gamma\left(\frac{\left(-\phi_2^2+\phi_3\right)\ln 2}{\phi_2^2}\right)}\left(\frac{\phi_2}{\ln 2}\right)^n n^{\left(-1+\frac{\phi_3}{{\phi_2}^2}\right)\ln 2-1}.$$

By constant variation we find that

$$c'(z)=\frac{a'(z)}{(\mathrm{e}^{\phi_2 z}-2)g(z)},$$

and therefore since $b(z)=c(z)g(z)$ we have

$$b(z)=g(z)\int\limits_0^z\frac{a'(t)}{(\mathrm{e}^{\phi_2 t}-2)g(t)}$$

From now on the same arguments hold as the ones we saw in the last part of Section 5.5 to conclude the equivalent result.

5.7 Asymptotics for general r

[Proof of Theorem 5.2.6].

In some applications it might be the case that we would like to allow a different kind of repetitions like exactly k repetitions at each iteration step, or a subset $\{3, 4, 8\}$, so that at each iteration step we could select 3 or 4 or 8 leaves to expand. In this case r is any non-empty subset of integers $r \subset \mathbb{N}^*$.

It is interesting to see that even in this case we can get a very general asymptotic formula for the coefficients of such processes. In fact, the proof is very close to the one presented in Section 5.8. The only difference is that the asymptotic formula is more general.

The initial condition of the process has to be little modified, since for example if we allow for exactly 2 repetitions at each iteration step (i.e $r = \{2\}$), we can not start with a single leaf, since at each iteration step we have to select exactly 2 leaves and only one is available. We will denote by $min(r)$ the smallest integer present in the set m. Therefore, the initial condition of the process is a single tree which is a leaf with $min(r)$ leaves attached to it. So it is always possible to iterate.

Finally, depending on the set r of allowed repetitions, some coefficients may always be 0. For instance if $r = \{2\}$, the number of trees of odd sizes is always 0. This why the asymptotics of the Theorem works for n of the form $n = 0(mod\, m)$.

It turns out that our analysis in Section 5.8 is robust, and that studying this case is just about adding some details to the analysis that we have already done.

The initial condition on the coefficient of B_n differs following to the minimal number of allowed repetitions. We will denote this number $m = min(r)$. Then $B_k = 0$ for all $k < m$ and $B_m = 1$. The recurrence goes on afterwards as described in Equation (5.2). The reason is that if number of allowed repetitions is at least 2 we can start with a single leaf. So we start with a single root containing 2 leaves.

If we do a ***Borel transform*** on the coefficients of B_n, we get the same equation as Equation (5.11), but the first two sums now range only over accepted values of r, since some repetitions might not be allowed.

$$b_n = \begin{cases} 0 & \text{if } n < m \\ \dfrac{1}{m!} & \text{if } n = m \\ \displaystyle\sum_{k=1,k\in r}^{\infty} \phi_2^k \binom{n-k}{k} \frac{(n-k)!}{n!} b_{n-k} & \\ \displaystyle + \sum_{k=2,(k-1)\in r}^{\infty} \phi_2^{k-2}\phi_3(k-1) \binom{n-k}{k-1} \frac{(n-k)!}{n!} b_{n-k} & \text{if } n > m \\ \displaystyle + \sum_{k=3}^{n-1} \left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2,u_i\neq 0}^{|occ(a)|} \phi_i^{u_i} \right) \frac{(n-k)!}{n!} b_{n-k} & \end{cases} \tag{5.15}$$

It is then possible to use Lemma 5.6.3 and Lemma 5.6.4 seen in Section 5.6.

In order to get the asymptotic equivalent wee that it is to write b_n as follows:

$$b_n = \sum_{k=1, k\in r}^{\infty} \frac{{\phi_2}^k}{k!}\left(1-\frac{k^2}{n}\right) b_{n-k} + \frac{\phi_3}{\phi_2^2} \sum_{k=2,(k-1)\in r}^{\infty} \frac{{\phi_2}^k}{k!}\left(\frac{k^2}{n}-\frac{k}{n}\right) b_{n-k} + a_n \qquad (5.16)$$

We can then show by using Lemma 5.6.7 that:

$$a_n = O\left(\frac{b_n}{n^2}\right),$$

It is true that this time b_n can be exponentially decreasing. This is where the condition of ϕ_n intervenes. By using Lemma 5.6.7 we see that

$$a_n \leq \sum_{k=1}^{n-3} \frac{c}{m!^{k/m}\, n^3} b_{n-k}$$

We know that min$(r) = m$ but $r \neq m$, since this case has been treated previously. This implies that the singularity of $b(z)$ is smaller than $\frac{m!^{1/m}}{\phi_2}$. Let $g(z) = \sum_{k=1,k\in r}^{\infty} \frac{{\phi_2}^k z^k}{k!}$ we have,

$$\frac{1}{z} \sum_{k=1, k\in r}^{\infty} \frac{{\phi_2}^k (k-k(k-1)) z^k}{k!} = g'(z) - z g''(z).$$

and,

$$\frac{\phi_3}{{\phi_2}^2 z} \sum_{k=2,(k-1)\in r}^{\infty} \frac{{\phi_2}^k k\,(k-1)\, z^k}{k!} = \frac{\phi_3\, z}{\phi_2} g'(z).$$

Then by coefficient translation of Equation (5.16) we get

$$\left(\left(1+\frac{\phi_3\, z}{\phi_2}\right) g'(z) - z\, g''(z)\right) b(z) + (g(z)-1)\, b'(z) + a'(z) = 0$$

by solving the homogeneous equation we obtain

$$r(z) = \exp\left(\int_0^z \frac{(1+\frac{\phi_3\, z}{\phi_2}) g'(z) - z\, g''(z)}{g(z)-1}\, dz\right)$$

We see that the real dominant singularity ζ is the smallest real solution of $g(z) - 1 = 0$ Let $d_1(z) = \frac{(1+\frac{\phi_3\, z}{\phi_2})g'(z) - z\, g''(z)}{g(z)-1}$. Then

$$\begin{aligned}\lim_{z\to\zeta} (z-\zeta)\, d_1(z) &= \lim_{z\to\zeta}\left(\frac{(1+\frac{\phi_3\, z}{\phi_2})g'(z) - z\, g''(z)}{g'(z)(z-\zeta)}\right)(z-\zeta)\\ &= 1 + \frac{\phi_3\,\zeta}{\phi_2} - \frac{\zeta\, g''(\zeta)}{g'(\zeta)}.\end{aligned}$$

The asymptotic behaviour of $r(z)$ around ζ mechanically obtained to be

$$r(z) \underset{z\to\zeta}{=} h(z-\zeta)\ (z-\zeta)^{-1-\frac{\phi_3\,\zeta}{\phi_2}+\frac{\zeta\, g''(\zeta)}{g'(\zeta)}}$$

for some function h that is analytic at 0 and $h(0) \neq 0$. Then

$$r_n \underset{n\to\infty}{=} \kappa \left(\frac{1}{\zeta}\right)^n n^{-1+\frac{\zeta\,\phi_3}{\phi_2}-\frac{\zeta\,g''(\zeta)}{g'(\zeta)}}$$

From there the result follows. Then it is possible to write

$$b(z) = c(z)r(z), \quad c(z) = \int\limits_0^z \frac{a'(t)}{(g(t)-1)\,r(t)}$$

The rest of the calculations follow by using the method used in the end of Section 5.5.

Finally, to obtain in the asymptotic form given by Theorem 5.2.6, the function $f(z)$ is defined by $f(z) = \sum\limits_{i=1, i\in r}^{\infty} \frac{z^i}{i!}$ while in the proof that we did we used $g(z)$ that is defined as $g(z) = \sum\limits_{i=1, i\in r}^{\infty} \frac{(\phi_2\,z)^i}{i!}$.

We see that, the solutions ζ_i of $g(z) - 1$ have the form $\zeta_i = \rho_i/\phi_2$ while the solutions of $f(z) - 1$ are just the ρ_i. Now in the proof ζ was the singularity of smallest modulus of $g(z) - 1$. Then

$$\rho = \phi_2\,\zeta,$$

is the singularity of smallest modulus of $f(z) - 1$. Therefore, it suffices to replace ζ by ρ/ϕ_2 and also $g''(\zeta)/g'(\zeta) = \phi_2 f'(\rho)/f''(\rho)$ in the asymptotics of b_n to get the exact form.

5.8 Asymptotic analysis for $r = [m]$

[Proof of Theorem 5.2.8]. The proof of this Theorem is the same as the one for Theorem 5.2.6. The only difference is in the treatment of δ_1.

Theorem 5.2.8 also shows us how the value of the singularity moves from $\frac{1}{\phi_2}$ to $\frac{\ln 2}{\phi_2}$ as we allow more leaves to evolve in the same time.

The transcendental ρ when $r = \mathbb{N}^*$ is a rare phenomena in the asymptotic form arising from combinatorial problems. It reflects the fact that the specification of the problem is an infinite sum in this case or equivalently it is based on a particular substitution see Section 5.6.

This time the function $g(z)$ seen in **??** contains all the coefficients :

$$g(z) = \sum_{k=1}^{m} \frac{{\phi_2}^k z^k}{k!}. \tag{5.17}$$

From **??** we have:

$$\begin{aligned}\delta_1 &= \lim_{z\to\zeta} (z-\zeta)\,d_1(z)\\ &= \lim_{z\to\zeta} \frac{(1+\frac{\phi_3\,z}{\phi_2})\phi_2\left(f(z)+1-\frac{(\phi_2\,z)^m}{m!}\right) - z\,\phi_2\left(\phi_2\left(f(z)+1-\frac{(\phi_2\,z)^{m-1}}{(m-1)!}-\frac{(\phi_2\,z)^m}{m!}\right)\right)(z-\zeta)}{\phi_2\left(f(z)+1-\frac{(\phi_2\,z)^m}{m!}\right)(z-\zeta)}\end{aligned}$$

$$
\begin{aligned}
&= \frac{(1+\frac{\phi_3\,\zeta}{\phi_2})\phi_2\left(f(\zeta)+1-\frac{(\phi_2\,\zeta)^m}{m!}\right)-\zeta\,\phi_2\left(\phi_2\left(f(\zeta)+1-\frac{(\phi_2\,\zeta)^{m-1}}{(m-1)!}-\frac{(\phi_2\,\zeta)^m}{m!}\right)\right)}{\phi_2\left(f(\zeta)+1-\frac{(\phi_2\,\zeta)^m}{m!}\right)}\\
&= 1+\frac{\phi_3}{\phi_2}\,\zeta-\phi_2\,\zeta+\frac{(\phi_2\,\zeta)^m\;\phi_2}{(m-1)!\,f'(\zeta)}\\
&= 1+\tau+\phi_2\,\zeta\left(-1+\frac{\phi_3}{\phi_2^2}\right).
\end{aligned}
$$

By letting $\beta=\frac{\phi_2(\phi_2\,\zeta)^m}{(m-1)!f'(\zeta)}$ and $\rho=\phi_2\,\zeta$. We make the same shift as in the end of the previous section. We finally get $\tau=\frac{\rho^m}{(m-1)!f'(\rho)}$ as defined in Theorem 5.2.8.

5.9 Asymptotics when unary nodes are allowed

In this section, we see how allowing unary nodes affects the study of evolution processes. An important question directly related to this problem is how is it possible to allow unary nodes in the *evolution processes* (see Section 5.1 and Definition 5.2.1) that we have been studying?

In fact, it is possible but we have to add a little constraint on the *evolution process*. We recall that the processes that we study have a notion size and that this notion size for us is the number of individuals that can develop in the next iteration step. On the level of trees, this corresponds to the number of leaves of the tree. If we allow unary nodes in all generality. so that at each iteration step all leaves that are evolving make unary nodes. The overall size of the tree does not change, and as a consequence, there is an infinite number of trees for each fixed size.

Therefore, it is possible to allow unary nodes, but we have to be careful. In fact, it is impossible to allow unary nodes if the set of allowed repetitions is equal to one (i.e $r=\{1\}$). We will need this set to be different from $\{1\}$. We require the following additional constraint, at least one of the leaves evolving at an iteration step has to evolve into a non-unary node. This means that all other evolving leaves can make unary nodes.

This also implies that the *weighted degree function* can not contain only unary nodes.

The recurrence on the coefficients of this model does not change Equation (5.2), we give it again:

$$
\begin{cases}
B_1=1,\\
B_n=\displaystyle\sum_{k=1}^{n-1}\left(\sum_{a\in A_{n,k,r,\phi}}\frac{(n-k)!}{(n-k-|a|)!}\prod_{i=0,u_i\neq 0}^{|occ(a)|}\frac{\phi_{i+1}^{u_i}}{u_i!}\right)B_{n-k}
\end{cases}
$$

We write B_n instead of $B_n^{r,\phi}$ and define,

$$
\begin{aligned}
&s_1=1\\
&s_n=\phi_2\;(n-1)\left(\sum_{i=1,i\in r}^{\infty}\phi_1^{i-1}\binom{n-1}{i-1}\right)s_{n-1},\qquad \text{for } n>1.
\end{aligned}
\tag{5.18}
$$

We immediately find have,

$$s_n = \phi_2^{n-1} \prod_{k=1}^{n-1} (n-k) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-k-1}{i-1} \right).$$

The idea now is to renormalize B_n, for all $n \geq 1$,

$$b_n = \frac{B_n}{\phi_2^{n-1} \prod_{t=1}^{n-1} (n-t) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-t-1}{i-1} \right)}$$

We get a new recurrence,

$$\begin{aligned} b_1 =& 1 \\ b_n =& \sum_{k=1}^{n-1} \left(\Bigg(\sum_{a \in A_{n,k,r,\phi}} \frac{(n-k)!}{(n-k-|a|)!} \prod_{i=0, u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!} \Bigg) \right. \\ & \left. \cdot \frac{\phi_2^{n-1-k} \prod_{t=k+1}^{n-1} (n-t) \Big(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-t-1}{i-1} \Big)}{\phi_2^{n-1} \prod_{t=1}^{n-1} (n-t) \Big(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-t-1}{i-1} \Big)} \right) b_{n-k} \end{aligned} \tag{5.19}$$

Notation. We will denote $c_{n,k}$ for the coefficient in front of b_{n-k}.
By analysing the coefficient of $c_{n,k}$ at $k = 1$, the possibilities for a tree of size $n-1$ to evolve into a tree of size n are

$$\phi_2(n-1) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-2}{i-1} \right).$$

Therefore,

$$\begin{aligned} &\phi_2(n-1) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-2}{i-1} \right) \left(\phi_2^{n-2} \prod_{t=2}^{n-1} (n-t) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-t-1}{i-1} \right) \right) \\ &= \phi_2^{n-1} \prod_{t=1}^{n-1} (n-t) \left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1} \binom{n-t-1}{i-1} \right). \end{aligned}$$

So that,

$$c_{n,1} = 1.$$

From the above discussion we see that,

$$b_n = b_{n-1} + \epsilon_n,$$

where ϵ_n groups all the of terms in Equation (5.19). Since they are all positive, we get immediately:

Lemma 5.9.1. *The sequence b_n is increasing, that is for all $n \geq 2$,*

$$b_n \geq b_{n-1}.$$

Proof. The proof is straightforward from the fact that $b_n = b_{n-1} + \ldots$ □

Now, if we are able to show that b_n is bounded to finish the proof. It would mean that:

$$b_n \underset{n\to\infty}{\sim} c,$$

for some constant c and therefore,

$$B_n \underset{n\to\infty}{\sim} c\, s_n.$$

This is where we want to go. Let us take some upper bounds on $c_{n,k}$.

$$\begin{aligned}
c_{n,k} \le{}& \Bigg(\sum_{i=1,i\in m}^{\infty} \phi_1^{i-1}\binom{n-k-1}{i-1}\Bigg)\Bigg(\Big(\sum_{a\in A_{n,k,r,\hat{\phi}}} \frac{(n-k)!}{(n-k-|a|)!}\prod_{i=0,u_i\neq 0}^{|occ(a)|}\frac{\phi_{i+1}^{u_i}}{u_i!}\Big)\\
&\cdot \frac{\phi_2^{n-1-k}\prod_{t=k+1}^{n-1}(n-t)\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}{\phi_2^{n-1}\prod_{t=1}^{n-1}(n-t)\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}\Bigg)\\
={}& \Bigg(\sum_{i=1,i\in m}^{\infty} \phi_1^{i-1}\binom{n-k-1}{i-1}\Bigg)\Bigg(\Big(\sum_{a\in A_{n,k,r,\hat{\phi}}} \frac{(n-k)!}{(n-k-|a|)!}\prod_{i=0,u_i\neq 0}^{|occ(a)|}\frac{\phi_{i+1}^{u_i}}{u_i!}\Big)\\
&\cdot \frac{\phi_2^{n-1-k}\,(n-k-1)!\prod_{t=k+1}^{n-1}\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}{\phi_2^{n-1}\,(n-1)!\prod_{t=1}^{n-1}\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}\Bigg)\\
={}& \Bigg(\Big(\sum_{a\in A_{n,k,r,\hat{\phi}}} \frac{(n-k)!}{(n-k-|a|)!}\prod_{i=0,u_i\neq 0}^{|occ(a)|}\frac{\phi_{i+1}^{u_i}}{u_i!}\Big)\\
&\cdot \frac{\phi_2^{n-1-k}\,(n-k-1)!\prod_{t=k}^{n-1}\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}{\phi_2^{n-1}\,(n-1)!\prod_{t=1}^{n-1}\Big(\sum_{i=1,i\in m}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\Big)}\Bigg)
\end{aligned}$$

where $\hat{\phi}(z) = \phi(z) - \phi_1 z$. The first inequality can be justified combinatorially, for each coefficient B_{n-k}, by expanding a subset of leaves into non-unary nodes, we should take this into account to allow some others to make unary nodes. But as an upper bound we consider the size of the subset of leaves that has expanded into non-unary nodes to be equal to 1, and allow all other leaves to unary nodes. Then in the first equality we pull out a term from the product (from both numerator and denominator). The last equality comes from inserting the sum $\left(\sum_{i=1,i\in r}^{\infty}\phi_1^{i-1}\binom{n-k-1}{i-1}\right)$ into $\prod_{t=k+1}^{n-1}\left(\sum_{i=1,i\in r}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)$.

Lemma 5.9.2. *For* $3 \leq k \leq n-1$*,*

$$\frac{\prod_{t=k}^{n-1}\left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1}\binom{n-t-1}{i-1}\right)}{\prod_{t=1}^{n-1}\left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1}\binom{n-t-1}{i-1}\right)} \underset{n \to \infty}{=} O\left(\frac{1}{n^{k-1}}\right).$$

Proof. It comes from the fact that $r \neq \{1\}$ and $r \neq \emptyset$. Therefore, the sum

$$\sum_{t=1}^{n-1}\left(\sum_{i=1, i \in r}^{\infty} \phi_1^{i-1}\binom{n-t-1}{i-1}\right),$$

is at least linear in n for each term with $1 \leq t \leq k-1$. □

Let us first denote

$$q_{n,k} = \Big(\sum_{a \in A_{n,k,r,\hat{\phi}}} \frac{(n-k)!}{(n-k-|a|)!} \prod_{i=0, u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!}\Big) \frac{(n-k)!}{n!}.$$

Where $\hat{\phi}$ has no unary-nodes. A remark is that,

$$q_{n,k} = \frac{t_{n,k}(n-k)!}{n!},$$

where $t_{n,k}$ has the same form as in Section 5.6.

$$\begin{aligned} t_{n,k} &= \sum_{a \in C_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i} \\ &= \sum_{a \in A_{n,k,\phi}} \frac{(n-k)!}{(n-k-|a|+1)!} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!} \end{aligned}$$

Lemma 5.9.3. *When* n *grows,*

$$q_{n,k} = O(1).$$

Proof. From Section 5.6, for $3 \leq k \leq n-1$, we have seen that,

$$\left(\sum_{a \in \mathbf{C}_{n,n-k,r,\phi}} \prod_{i=2, u_i \neq 0}^{|occ(a)|} \phi_i^{u_i}\right) \frac{(n-k)!}{n!} = O\left(\frac{1}{n^2}\right), \tag{5.20}$$

where $\mathbf{C}_{n,n-k,r,\phi}$ is $C_{n,n-k,r,\phi}$ with some removed configurations namely the ones that make only binary nodes or the ones that make a single ternary with the others all binary. See Remark 5.6.2 for more details. We can the proceed as in Lemma 5.6.6. But this time,

$$|C_{n,n-k,r,\phi}| \leq \binom{n-1}{k}.$$

And following the same steps we find that,

$$q_{n,k} = t_{n,k}\frac{(n-k)!}{n!} = O\left(1\right).$$

□

Now, we are ready, to give upper bounds on the coefficients $c_{n,k}$.

Lemma 5.9.4.

$$c_{n,2} \underset{n\to\infty}{=} O\left(\frac{1}{n^2}\right), \quad c_{n,3} \underset{n\to\infty}{=} O\left(\frac{1}{n^2}\right),$$

and for $4 \le k \le n-1$

$$c_{n,k} \underset{n\to\infty}{=} O\left(\frac{1}{n^{k-1}}\right).$$

Proof.

$$c_{n,k} \le t_{n,k}\left(\frac{n}{(n-k)}\right)\frac{\phi_2^{n-1-k}\prod\limits_{t=k}^{n-1}\left(\sum\limits_{i=1,i\in r}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)}{\phi_2^{n-1}\prod\limits_{t=1}^{n-1}\left(\sum\limits_{i=1,i\in r}^{\infty}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)} = O\left(\frac{1}{n^{k-1}}\right).$$

The result comes by combining Lemma 5.9.2 and Lemma 5.9.3

By a direct inspection of of $c_{n,2}$,

$$c_{n,2} \le \Bigg(\Bigg(\phi_2^2\binom{n-2}{2} + \phi_3\binom{n-2}{1} + \phi_1\phi_3(n-2)(n-3)\Bigg) \cdot \frac{\phi_2^{n-1-k}\prod\limits_{t=3}^{n-1}(n-t)\left(\sum\limits_{i=1,i\in m}^{max(m)}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)}{\phi_2^{n-1}\prod\limits_{t=1}^{n-1}(n-t)\left(\sum\limits_{i=1,i\in m}^{max(m)}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)}\Bigg)$$

We have,

$$\frac{\phi_2^{n-1-k}\prod\limits_{t=3}^{n-1}(n-t)\left(\sum\limits_{i=1,i\in m}^{max(m)}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)}{\phi_2^{n-1}\prod\limits_{t=1}^{n-1}(n-t)\left(\sum\limits_{i=1,i\in m}^{max(m)}\phi_1^{i-1}\binom{n-t-1}{i-1}\right)} = O\left(\frac{1}{n^4}\right).$$

Finally,

$$c_{n,2} \underset{n\to\infty}{=} O\left(\frac{1}{n^2}\right).$$

The same thing can be done for $c_{n,3}$,

$$c_{n,3} \underset{n\to\infty}{=} O\left(\frac{1}{n^2}\right).$$

□

If we sum up all the preceding we have,

$$\begin{aligned} b_n &= b_{n-1} + \sum_{k=2}^{n-1} c_{n,k}\, b_{n-k} \\ &\le b_{n-1} + b_{n-2} \sum_{k=2}^{n-1} c_{n,k} \\ &= b_{n-1} + b_{n-2}\left(\frac{c}{n^2}\right), \end{aligned} \tag{5.21}$$

for some positive constant c. It comes from

$$\begin{aligned} \sum_{k=2}^{n-1} c_{n,k} &= O\left(\frac{1}{n^2}\right) + \sum_{k=4}^{n-1} O\left(\frac{1}{n^{k-1}}\right) \\ &= O\left(\frac{1}{n^2}\right) \end{aligned}$$

Therefore, we can write a new recurrence,

$$\begin{cases} g_1 = 1, & \\ g_2 = \phi_2, & \\ g_n = g_{n-1} + \frac{c}{n^2}\, g_{n-2}, & \text{for } n > 2. \end{cases} \tag{5.22}$$

Lemma 5.9.5. *For all $n \ge 1$,*

$$g_n \ge b_n.$$

Proof. The proof follows Lemma 5.9.4, and the resulting upper bound on b_n in Equation (5.21). □

Let the $g(z)$ be the *generating function* of the sequence g_n:

$$g(z) = \sum_{n\ge 0} g_n z^n$$

It is possible to write a differential equation satisfying $g(z)$, we find,

$$-cg\,(z) - 2\,\partial g\,(z) + (1-z)\,\partial^2 g\,(z)\,,$$

with initial conditions that depend on the set r. We see directly that $z = 1$ is a regular singularity. The differential equation also satisfies the conditions of Theorem 2.4.18.

$$\begin{aligned} \delta_1 &= \lim_{z\to\zeta}(z-\zeta)a_1(z) \\ &= \lim_{z\to 1} \frac{2}{(z-1)}(z-1) = 2. \end{aligned}$$

Therefore the singular expansion is of the form,

$$(z-1)^{-1}\, H(z-1)\; + H_0(z-1)\,(\log(z-1))^k, \quad \text{with} \quad k \in \{0,1\},$$

where the functions $H(0)$ and $H_0(0)$ are different from 0. Therefore,

$$g_n \underset{n\to\infty}{\sim} c_2,$$

where c_2 is some constant that depends on the function $H(z)$. This concludes the proof of Theorem 5.2.7.

In fact it is possible to extend the Theorem to cases where $\phi_1 \geq 1$ and for some $j > 1$, $\phi_j \geq 1$. The same type of Theorem holds but with a shift on the coefficients since there appear some periodicities. In the following we give the statement of the Theorem without proof, because of the heavier notations and the fact that the proof is the same as the one that we already did.

Condition 5.9.6. *Let $\phi(z)$ be a weighted degree function as presented in Section 3.5.1 and such that $\phi_1 \geq 1$, and for some $d \geq 2$, $\phi_d \geq 1$ and $\phi_n = O\left(\frac{n!}{n^5}\right)$.*

Theorem 5.9.7. *Let $\phi(z)$ be as in Condition 5.9.6, and let $r \subset \mathbb{N}^*$, $r \neq \emptyset$, and $r \neq \{1\}$, then when n is of the form $n = 1\, mod\,(d-1)$,*

$$B_n^{r,\phi} \underset{n\to\infty}{\sim} \kappa \prod_{k=1,k=1mod\,(d-1)}^{n-1} (n-k)\left(\sum_{i=1,i\in r}^{\infty} \phi_1^{i-1}\binom{n-k-1}{i-1}\right),$$

where κ is a constant that depends on $\phi(z)$ and r.

5.10 Asymptotics where no binary nodes are allowed

So far we have been studying the *evolution process* (Definition 5.2.1) in terms of counting and asymptotic enumeration. The two main theorems that encompasses the other results namely Theorem 5.2.6 and Theorem 5.2.7. The Theorems work when we include binary nodes (i.e $\phi_2 \geq 1$).

We have also seen in Section 5.9, that for Theorem 5.2.7, it is possible to extend the result very naturally for situations where we do not have binary nodes but there exists some $j > 2$, such that $\phi_j \geq 1$ in Condition 5.9.6.

For cases where there are nor either unary nodes and nor binary nodes more diverse asymptotic situations appear, there is still much work to do to figure out the form of the asymptotics. We have resumed some results in Table 5.10.

There is a paper [**BGGW20**], in which the authors studied the cases when $r = \mathbb{N}^*$ and $\phi(z) = z^d$ for $d \geq 2$. Their main Theorem gives:

Theorem 5.10.1. *Let $r = \mathbb{N}^*$, and $\phi(z) = z^d$, then the class strictly monotonic Schröder tree $\mathcal{G}^d$, is specified by:*

$$G^d(z) = z + G^d(z + z^d) - G^d(z).$$

When n grows we get:

$$G_n^d \begin{cases} = \quad 0 & \textit{if}\, n \not\equiv 1 mod k-1, \\ \underset{n\to\infty}{\sim} \quad \eta_d\,(m-1)!\left(\dfrac{d-1}{\ln 2}\right)^m m^{\frac{2-d\ln 2}{2(d-1)}} & \textit{if}\, n = 1+(d-1)m. \end{cases} \tag{5.23}$$

We see from the Theorem that the form of the asymptotic behaviour has similarities with Corollary 5.2.10. It also reduces to it when $d = 2$.

For $\phi(z) = \phi_d z^d$ with $\phi_d \geq 2$, it is also quiet simple to give the asymptotics for when $r = \{1\}$. Since the recurrence reduces to

$$\begin{cases} B_1^d = 1, \\ B_n^d = \phi_d \, B_{n-d+1} \end{cases} \tag{5.24}$$

We will write B_n^d instead of $B^{\{1\},z^d}$ for simplicity Then, for $d > 2$, when n is of the form $n = 1 \, mod \, (d-1)$,

$$B_n^d = \phi_d^{\lfloor \frac{n}{d-1} \rfloor} \prod_{k=1}^{\lfloor \frac{n}{d-1} \rfloor} (n - (d-1)k) \, ,$$

We can define a new sequence

$$s_n^d = B_{n(d-1)+1}^d,$$

and then,

$$s_n^d = \phi_d^{n-1} \prod_{k=1}^{n-1} \Big((n-k-1)(d-1) + 1 \Big),$$

Then if $\phi_d = 1$, the *exponential generating function*(**EGF**) of $s(z)$

$$s^d(z) = (1 - (d-1)\, z)^{-\frac{1}{d-1}}.$$

From which by *singularity analysis*,

$$s_n^d \underset{n\to\infty}{\sim} \alpha \, n! \, (d-1)^n \; n^{-\frac{d-2}{d-1}}.$$

And then,

Theorem 5.10.2.
$$B_n^d \underset{n\to\infty}{\sim} \beta \left(\lfloor (\frac{n}{d-1}) \rfloor \right)! \; (d-1)^{\frac{n}{d-1}} \; n^{-\frac{d-2}{d-1}},$$

$\phi(z)$	r	Theorem	References
z^d	$\mathbb{N}^*$	Theorem 5.10.1	[**BGGW20**]
z^d	$\{1\}$	Theorem 5.10.2	
$\frac{z^d}{1-z}$	$\{1\}$	Conjecture 5.10.3	

Table 5.10: Some asymptotics when $\phi_2 = 0$ in the *evolution process* of Equation (5.1).

Open question (Stretched exponential). The parametrisation of Equation (5.1) is $r = \{1\}$, $d \geq 2$, and $\phi(z) = \frac{z^d}{1-z}$. Let $\mathcal{T}^d$ be the resulting class of trees. Then,[5]

$$T^d(z) = z + \frac{z^d}{1-z} \, \partial_z T^d(z).$$

The first values of T_n^d for $d = 2, 3, 4, 5, 6$ are written in Table 5.11. The sequences of T_n^2 and T_n^3 can be found in EIS A059480 and EIS A059480. $T^2(z)$ corresponds to the model of *increasing Schröder trees* studied in Section 4.2 and the enumeration problem is easily solved. However, for $d > 2$. The asymptotic enumeration is harder and it seems to involve

[5] $\partial_z = \frac{d}{dz}$ is the classical differential operator

T_n^2	T_n^3	T_n^4	T_n^5	T_n^6
1	1	1	1	1
1	0	0	0	0
3	1	0	0	0
12	1	1	0	0
60	4	1	1	0
360	8	1	1	1
2520	28	5	1	1
20160	76	10	1	1
181440	272	16	6	1
1814400	880	51	12	1
19958400	3328	131	19	7
239500800	12128	275	27	14
3113510400	48736	785	81	22
43589145600	194272	2226	201	31
653837184000	827840	5526	81	41
10461394944000	3547648	15731	410	118
177843714048000	15965248	46895	734	286

Table 5.11: The first values of $T_n^d, 2 \le d \le 6$ and $n \in \{1, .., 17\}$.

stretched exponential of the following form. The case where $d = 3$, can still be solved using exact different equations solutions. But for $d > 3$ the full asymptotic behaviour is still an open question. The recurrence has the form for a fixed $d \geq 2$,

$$\begin{cases} T_1^d & = 1, \\ T_d^d & = 1, \\ T_n^d & = 0, \qquad \text{for } 1 < n < d \\ T_n^d & = T_{n-1}^d + (n-d+1)\,T_{n-d+1}^d, \qquad \text{for } n > d. \end{cases} \tag{5.25}$$

Conjecture 5.10.3. *For $d \geq 3$, when n grows, we have,*

$$T_n^d \underset{n\to\infty}{\sim} \beta\left(\lfloor\frac{n}{d-1}\rfloor\right)!\ (d-1)^{\frac{n}{d-1}}\, \mathrm{e}^{\left(\sum_{k=1}^{d-2} c_k n^{\left(\frac{k}{d-1}\right)}\right)}\, n^{-1},$$

where c_k are rational constants.

We give this as a conjecture, as we were able to solve the case for $d = 3$ which is given below. The solution is based on an exact solution for the *generating function*. The cases where $d > 3$, can not be solved by differential equations since their orders are higher. We are able to get asymptotic expansions by a process of bootstrapping, however, this does not give a rigorous a proof. In Table 5.12, we give the expansions that we got for $d = 4$ and $d = 5$.

We show here how to solve the for $d = 3$. But we do not write all the details of the proof. The recurrence by Equation (5.25) gives:

$$\begin{cases} T_1^3 &= 1, \\ T_2^3 &= 0, \\ T_3^3 &= 1, \\ T_n^3 &= T_{n-1}^3 + (n-2)\, T_{n-2}^3, \quad \text{for } n > 3. \end{cases} \tag{5.26}$$

Applying a *Borel transform* on the recurrence gives a new recurrence that we call t_n, so that:

$$t_n = \frac{T_n^d}{n!}.$$

$$\begin{cases} t_1 &= 1, \\ t_2 &= 0, \\ t_3 &= 1, \\ t_n &= \frac{(n-1)}{n(n-1)} t_{n-1} + \frac{(n-2)}{n(n-1)} t_{n-2} \quad \text{for } n > 3. \end{cases} \tag{5.27}$$

From which it is possible to write a *differential equation* for the *generating function* $t(z)$ of the sequence t_n.

d	Asymptotics
3	$c_1 \left(\frac{n}{2}\right)!\, 2^{\frac{n}{2}}\, e^{n^{1/2}} n^{-1}$
4	$c_2 \left(\frac{n}{3}\right)!\, 3^{\frac{n}{3}} \left(e^{\frac{n^{2/3}}{2}+\frac{n^{1/3}}{6}}\right) n^{-1}$
5	$c_3 \left(\frac{n}{4}\right)!\, 4^{\frac{n}{4}} \left(e^{\frac{n^{3/4}}{3}+\frac{n^{1/2}}{8}+\frac{5\, n^{1/4}}{96}}\right) n^{-1}$

Table 5.12: Conjectured asymptotic behaviour for $d = 4$ and $d = 5$, the case for $d = 3$ is solved in this section

$$(-z-1)\, \partial_z\, t\,(z) + \partial_z^2\, t\,(z) + 1$$

with initial conditions $t\,(0) = 0$ and $\partial_z\,(t)\,(0) = 1$. Then we can solve this equation with a computer algebra system and find that:

$$t(z) = \int_0^z -\frac{\sqrt{2\pi}}{2} \operatorname{erf}\left(\frac{\sqrt{2}}{2}(x+1)\right) e^{\frac{(x+1)^2}{2}} + e^{\frac{x(x+2)}{2}} \left(\frac{\sqrt{2\pi}}{2} \operatorname{erf}\left(\frac{\sqrt{2}}{2}\right) e^{\frac{1}{2}} + 1\right) \mathrm{d}x,$$

where $\operatorname{erf}(z)$ is the error function which is defined for all complex z by:

$$\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2}\, \mathrm{d}t.$$

.So that, we have:

$$t'(z) = -\frac{\sqrt{2\pi}}{2} \operatorname{erf}\left(\frac{\sqrt{2}}{2}(z+1)\right) e^{\frac{(z+1)^2}{2}} + e^{\frac{z(z+2)}{2}} \left(\frac{\sqrt{2\pi}}{2} \operatorname{erf}\left(\frac{\sqrt{2}}{2}\right) e^{\frac{1}{2}} + 1\right).$$

We notice that this function is entire. An expansion of $t'(z)$ around $z = \infty$, yields:

$$t'(z) \underset{z\to\infty}{\sim} \left(-\frac{\sqrt{2\pi}}{2} e^{\frac{1}{2}} + \frac{\sqrt{2\pi}}{2} \operatorname{erf}\left(\frac{\sqrt{2}}{2}\right) e^{\frac{1}{2}} + 1\right) e^{\frac{z^2}{2}+z} + O\left(\frac{1}{z}\right).$$

Then by *saddle-point analysis*,

$$[z^n]t'(z) = \frac{t_{n+1}}{n!} \underset{n\to\infty}{\sim} \alpha \left(\left(\frac{n}{2}\right)!\, 2^{\frac{n}{2}}\right)^{-1} \mathrm{e}^{\sqrt{n}},$$

with $\alpha = 1/4\, \left(\sqrt{\pi}\sqrt{2}\mathrm{erf}\left(1/2\,\sqrt{2}\right)\mathrm{e}^{1/2} - \sqrt{\pi}\sqrt{2}\mathrm{e}^{1/2} + 2\right)\mathrm{e}^{-1/4}$.

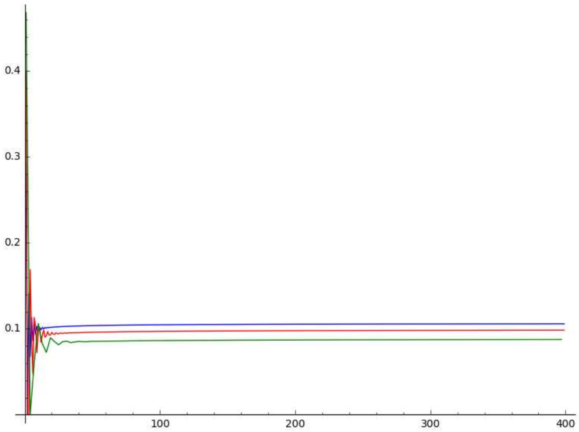


Figure 5.12: Simulation for $n \in \{1, 400\}$ of (Blue) T_n^3 plotted against its asymptotic behaviour. (Red) T_n^4 and (Green) T_n^5 are plotted against their conjectured behaviour in Table 5.12.

Finally, with a shift on the coefficients of $t'(z)$ we have,

$$[z^{n-1}]t'(z) = \frac{t_n}{(n-1)!} \underset{n\to\infty}{\sim} \beta\sqrt{n} \left(\left(\frac{n}{2}\right)!\, 2^{\frac{n}{2}}\right)^{-1} \mathrm{e}^{\sqrt{n}},$$

with $\beta = 1/4\, \left(\sqrt{\pi}\sqrt{2}\mathrm{erf}\left(1/2\,\sqrt{2}\right)\mathrm{e}^{1/2} - \sqrt{\pi}\sqrt{2}\mathrm{e}^{1/2} + 2\right)\mathrm{e}^{-1/4}$. Multiplying by $\frac{1}{n}$ we find,

$$\frac{t_n}{n!} \underset{n\to\infty}{\sim} \beta \frac{1}{\sqrt{n}} \left(\left(\frac{n}{2}\right)!\, 2^{\frac{n}{2}}\right)^{-1} \mathrm{e}^{\sqrt{n}},$$

We find finally after multiplying t_n by $n!$ that,

$$T_n^3 \underset{n\to\infty}{\sim} c\left(\frac{n}{2}\right)!\,\left(\sqrt{2}\right)^n \frac{\mathrm{e}^{\sqrt{n}}}{n},$$

with $c = 1/2\, \frac{\left(-\sqrt{\pi}\mathrm{e}^{1/2} + \sqrt{\pi}\mathrm{e}^{1/2}\mathrm{erf}\left(1/2\,\sqrt{2}\right) + \sqrt{2}\right)\mathrm{e}^{-1/4}}{\sqrt{\pi}} = 0.106979181603588\ldots$, which is in accordance with Conjecture 5.10.3.

Figure 5.12 shows how the asymptotic first order of T_n^d converges to the constant c.

5.11 Conclusion

In this chapter, we have seen how the *evolution process* we define, can be specialised to different classes of existing trees and several interesting new ones. We have been able to systematically enumerate and give the asymptotic behaviour of many models. However, our study falls short of giving the asymptotic behaviour of tree models where no binary nodes are allowed.

On the theoretical level, this chapter gives an interpretation for some transcendental polynomials terms that appeared in earlier research paper. On the other hand, we give a first study of trees with weakly increasing labellings along branches.

A lot of work has still to be done for cases without binary nodes, but we tried to give directions and conjectures for future studies about this tree classes.

CHAPTER 6

Average compaction of increasing tree models

Contents

> Kennst du das Land, wo die Zitronen blühn, Im dunklen Laub die Goldorangen glühn, Ein sanfter Wind vom blauen Himmel weht, Die Myrte still und hoch der Lorbeer steht? Kennst du es wohl? Dahin, dahin Möcht ich mit dir, o mein Geliebter, ziehn![1]
>
> Johann Wolfgang von Goethe (1749-1832), Kennst du das Land

[1] Do you know the land where the lemon-trees grow, In darkened leaves the gold-oranges glow, A soft wind blows from the pure blue sky, The myrtle stands mute, and the bay tree high? Do you know it well? It's there I'd be gone, To be there with you, O, my beloved one! Translated by azucarinho on lyricstranslate.com

6.1 Introduction

In this chapter we study the average compression rate of two tree classes. Compression of data structures is an important part in the study of data structures both on a theoretical and practical levels.

Tree-shape data structures are present in a lot of places in computer science. In compilation and parsing the syntax structure of a program is a tree, in algebra systems the symbolic expressions have a tree structure. XML documents and other markup languages use tree data structures.

Our focusing point of view is *lossless compression*, where the original data structure can be perfectly recovered from the compacted one. An introduction to the subject can be found in [**Say12**].

The compression of tree structures has been studied within different fields, such that computer science, information theory and combinatorics. The idea is that in a single tree, some subtrees can be isomorphic and therefore when compressing the tree we can keep only one occurrence of a repeated subtree and put pointers to it in the other occurrences which save an important amount of spacial memory. Usually an algorithmic step called the *common subexpression recognition* is run to identify identical fringe subtrees (*i.e.* a node and all its descendants) so that only one occurrence is stored and all other are replaced by pointers to the first one.

From the previous discussion, in computer science it is more common to talk about the size of the compacted tree while in combinatorics it is more common to talk about the number of non-isomorphic subtrees of a tree. The terms are different but they mean the same thing so we could use them interchangeably. Moreover, the size in terms of number nodes of the compacted tree, corresponds also to the number of non-isomorphic subtrees of this tree. In Figure 6.1 we have depicted a plane binary tree and its compacted version, we see that the compacted version has only 5 nodes while the original had 17 nodes. We also notice that the compacted tree has no longer a tree structure but rather a structure of a DAG (*directed acyclic graph*).

We will study the average compression rates of ***plane binary trees*** and ***Pólya tree*** (see Section 3.5 for their definitions) under what we call *increasingly labelled distribution*. We will see that this corresponds also exactly to the study of the average compression rate in *increasing binary trees* and *recursive trees*, when we do not consider the labels of the trees in the compression process.

Definition 6.1.1. *The* ***increasingly labelled distribution*** *over the unlabelled rooted tree class $\mathcal{T}$ corresponds to the uniform distribution of tree-shapes (trees obtained after removing all labels) in $\mathcal{IT}$, where $\mathcal{IT}$ is the increasing labelled version of $\mathcal{T}$.*

The tree-shapes in $\mathcal{IT}$ are exactly the trees in $\mathcal{T}$, but the probability distribution is different. Since the weight of its tree shape depends on its number of increasing labellings. Therefore, it leads to putting more weights on dense trees with small depths, since these trees have many different increasing labellings.

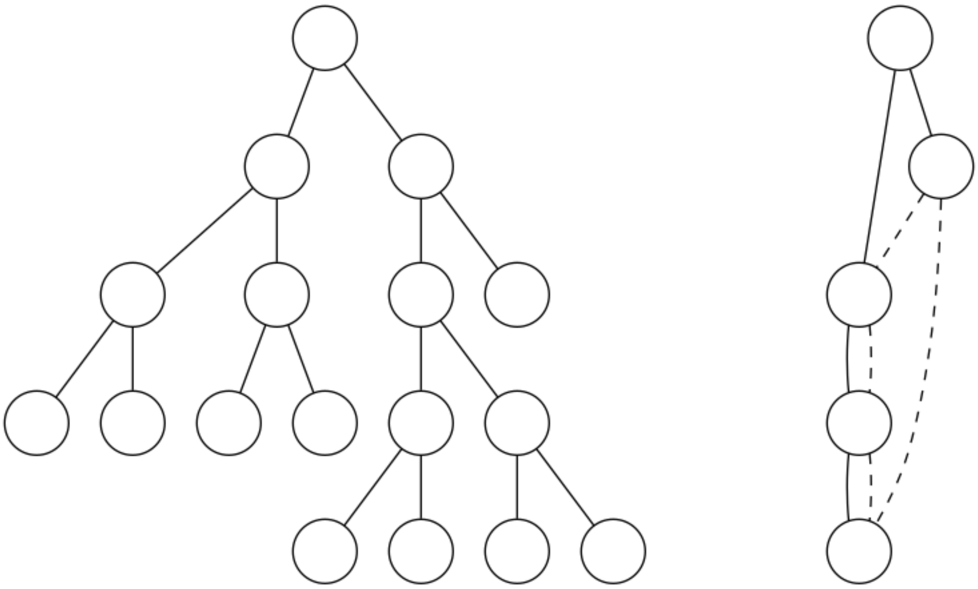


Figure 6.1: (left) A binary tree of size 17. (right) Its compacted version that has size 5.

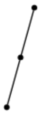
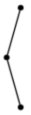
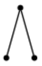
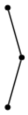
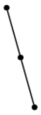
					
Uniform distribution	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$
Increasingly labelled distribution	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{1}{6}$	$\frac{1}{6}$

Figure 6.2: In the increasingly labelled distribution a tree gets a weight proportional to the number of ways to increasingly label it.

Sampling from the **increasingly labelled distribution** is equivalently done by sampling a tree uniformly at random in $\mathcal{IT}$ and then erasing its labels. In Figure 6.2 we draw plane binary trees of size 3 with their distribution according to both uniform and increasingly labelled distributions. If we let X_n be a random plane binary tree of size n sampled according to the increasingly labelled distribution. Then

$$\mathbb{P}(X_n = t) = \frac{\ell(t)}{IB_n},$$

where $\ell(t)$ denotes the number of ways to increasingly label the tree-shape t and IB_n is the total number of increasing binary trees of size n.

There, a random plane binary tree under *increasingly labelled distribution* can be sampled by taking a tree uniformly at random from *increasing binary trees* (also *binary search trees*) and erasing its labels. See Section 3.5.5, for a discussion on building a uniform tree in classes of increasing trees.

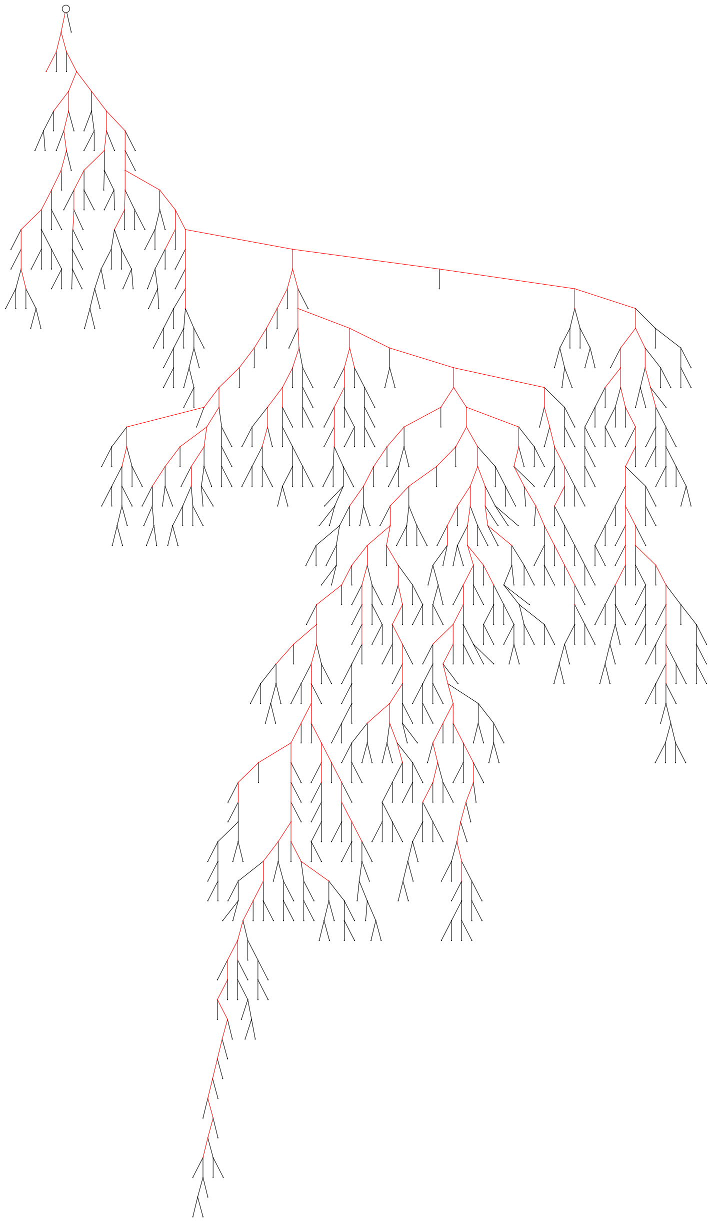


Figure 6.3: A uniformly sampled plane binary tree with 500 internal nodes: black fringe subtrees are removed by the compaction process; the red head is of size 250.

Another example, is the one of *Pólya* trees, a random tree under*increasingly labelled distribution* corresponds to uniformly picking a tree in the class of *recursive trees*.

One of the main parameter of interest in information theory is the entropy of the data structure: it represents an optimal lower bound on the average number of bits required to represent the data structure: see for example [**CT05**] for an introduction to the subject. For trees, the entropy of some models of plane trees have been studied in particular in [CMST17, GMS18, MTS18].

An analysis of a model of non-plane binary trees has been presented in [CMST17]. The authors focus on the number of symmetry nodes (internal nodes having two isomorphic subtrees as children) and its relation with Rényi entropy. As it has been said before, these questions are well defined when the probability distribution over the trees is defined. The last reference focuses on *increasingly labelled* (also binary search tree) distribution model. Likewise, it can be rephrased as the binary increasing tree model we will deal with in Section 6.3, as it was already pointed out in [BFS92].

A seminal paper of Flajolet *et al* [FSS90] considers the compaction ratio of binary trees under *uniform distribution*. They prove that the average size of the compacted result is $\alpha n / \sqrt{\log n}$ with a computable constant α. In the end of the paper the authors finally state that their analysis is fully adapted to all families of *simple trees* (see Section 3.5.1) under *uniform distribution*.

We recall that in the context of simple trees under *uniform distribution* of size n, the typical depth is of order $\sqrt{n}$ (this is the case for the binary trees). Bousquet-Mélou *et al.* [BMLMN15] present the complete proof for the compaction quantitative analysis of all varieties of simple trees and apply it experimentally on XML-trees. Finally, in [RW15] the authors are interested in the number of fringe subtrees with at least r occurrences in a random *simple tree* under *uniform distribution*. This approach is an extension of the previous results where it was dealt with subtrees appearing at least once (thus for $r = 1$).

As we have seen, *simple trees* under *uniform distribution* models have been well studied in terms of compression rate. However, not much has been done for these trees under other distributions and particularly under the *increasingly labelled distribution*. This distribution is particularly interesting, because it includes other models like *random search trees* and additionally these trees have a different shape on average, and their depth is typically of order $\log n$ in contrast with $\sqrt{n}$ under the uniform model.

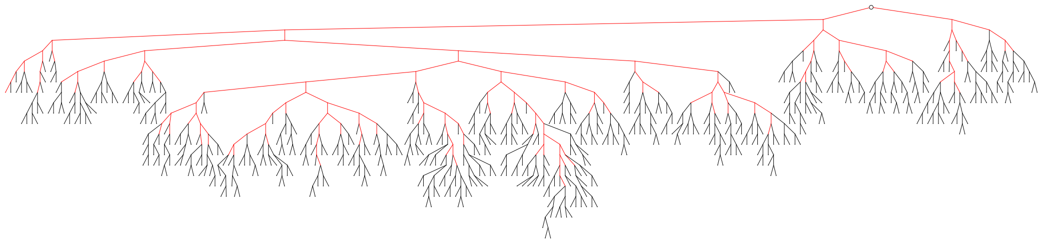


Figure 6.4: A uniformly sampled (plane) binary search tree structure with 500 internal nodes: black fringe subtrees are removed by the compaction; the red head is only of size 172

In Figure 6.3 we have represented a uniformly sampled binary tree with 500 internal nodes. If we compact it then all the fringe subtrees in black are removed and only the red structure is kept with addition of several pointers (that are not represented in the figure). The remaining red tree is of size 250. In contrast Figure 6.4 shows a binary search tree structure of size

500, with 172 nodes (represented in red) remaining after compaction. The gain compaction between the two structures is quite important.

Our study focuses on the number of non-isomorphic subtrees in a tree and this corresponds also to the size of the compacted tree (also called minimal DAG representation in [**ZYK13**]). This parameter is different from the study of symmetry nodes mentioned above (see [**CMST17**]), since there symmetries happen if an internal node has two isomorphic children (a local symmetry) whereas the number of non-isomorphic subtrees of a tree is capturing a global symmetry. Using the results in [**CMST17**] to design and analyse a data compression algorithm leads to constant compression rate on average, as was already shown in [**FGM97**]. In our case, we gain on average at least a logarithmic factor.

The distribution on plane binary trees we use is the same as the one of [**CMST17, MTS18**]. Even if the analysed parameters are not the same, the mathematical tools are based for all such studies on differential equation analyses due to the underlying distribution on trees.

We are interested in the analysis of the compaction ratio, relating the tree-size and its minimal DAG size as in [**ZYK13**].

Section 6.2 is dedicated to the study of the average compression ratio of Pólya trees under the *increasingly labelled* distribution defined in Definition 6.1.1 which corresponds to *unlabelled recursive trees*. Then, in Section 6.3, we study the average compaction ratio of binary trees under the same distribution, which corresponds to *unlabelled binary search trees*.

Both these families have been much studied lately in both probabilities theory [**Drm03, BDMdlS08, DIMR09, SM95**] and in combinatorics [**BFS92, KP07, PP07**].

Our main contribution:

For *binary trees* under *increasingly labelled distribution*, we prove that, asymptotically, if a random tree of size n denoted X_n is compacted, then the resulting structure has an average size of

$$\mathbb{E}(X_n) = \Theta\left(\frac{n}{\ln n}\right).$$

For *Pólya trees* under *increasingly labelled distribution*, we prove that, asymptotically, if a random tree of size n denoted Y_n is compacted, then the resulting structure has an average size of

$$C_1\sqrt{n} \leq \mathbb{E}(Y_n) \leq C_2\frac{n}{\ln n}.$$

For some constants C_1 and C_2.

We thus remark that such kind of trees are compacted in a more efficient way (in the sense of the number of remaining nodes) than the same models (plane binary tree and Pólya trees) under *uniform distribution*.

In Section 6.4, we present a new lossless data structure based on the compaction of binary search trees (bst). An experimental study is provided by using a prototype in *python* for our new data structure, the *compacted* bst. The experiments are very encouraging for the development of such new compacted search tree structures.

We conclude this chapter in Section 6.5 with a discussion about the reason why we were not able to show a stronger result. The discussion leads us to make to formulate conjectures and open questions for future works.

6.2 Average compression of Pólya trees under increasingly labelled distribution

The class of recursive trees has been studied by Meir and Moon [**MM78**] (see Section 3.5.4 for more details). These trees are models in several contexts as e.g. for the study of epidemic spreads, and thus many quantitative study have focused on this family. Some details are presented either in [**Drm09**] or in [**FS09**]. Using the classical operators from Analytic Combinatorics presented Section 2.2, recursive trees can be specified by the boxed product (also called Greene operator) seen in Section 2.2.2,

$$\mathcal{T} = \mathcal{Z}^{\square} \star \operatorname{Set}(\mathcal{T}), \tag{6.1}$$

meaning that the structure of a recursive tree (in the class $\mathcal{T}$) is defined as a root $\mathcal{Z}$ attached to a set of recursive trees (the set may be empty, then $\mathcal{Z}$ is a leaf) and such that the whole structure is canonically labelled (1,2,…, up to the size). The box in the boxed product indicates that the lowest label goes into the left component (the atom in this case). The atoms $\mathcal{Z}$ in the structure are therefore labelled increasingly on each path from the root of the tree to any leaf. In Figure 6.5 we have represented a recursive tree structure containing 5,000 nodes on the left-hand side. It has been uniformly sampled among all trees with the same size. The original root of the tree is represented using a small circle ○. On the right-hand side we have

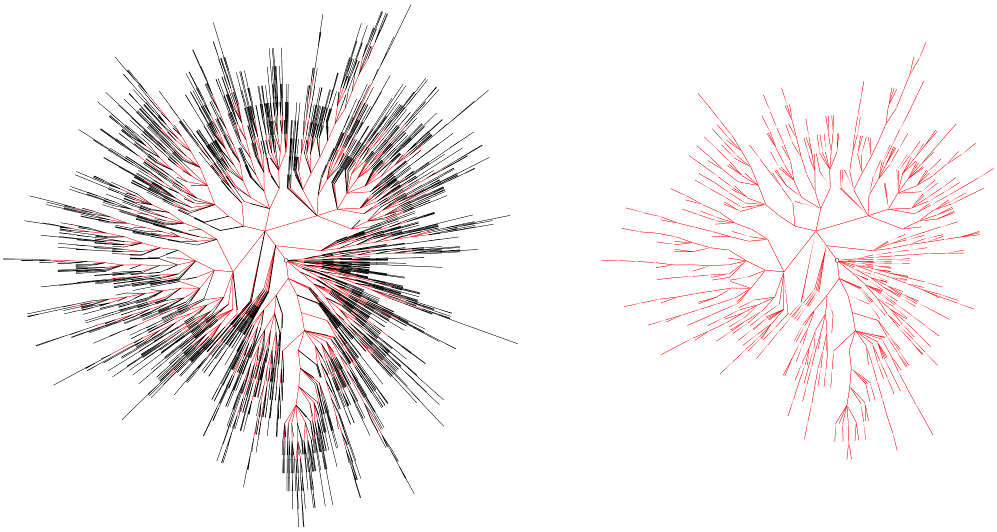


Figure 6.5: (left) a uniformly sampled non-plane recursive tree of size 5,000: black fringe subtrees are removed by the compaction; (right) the red head is of size 663

depicted the nodes that are kept after the compaction of the latter tree. There are only 663 nodes remaining.

Let the **EGF** of $T(z)$ be:

$$T(z) = \sum_{n \geq 1} T_n \frac{z^n}{n!},$$

where T_n corresponds to the number of trees containing n nodes *i.e.* of *size* n. We get directly from the specification Equation (6.1) that:

$$T(z) = \int_0^z \exp(T(v))\,\mathrm{d}v.$$

The unique power series solution satisfying $T(0) = 0$ is

$$T(z) = \ln \frac{1}{1-z},$$

whose dominant singularity is $\rho = 1$. Finally, we get the value $T_n = (n-1)!$.

Let $\mathcal{T}_n$ be the class of recursive trees of size n; the size of a tree τ is defined as the number of its nodes and is denoted by $|\tau|$. Let X_n be the size of the compacted tree corresponding to a random recursive tree τ of size n. In other words, X_n is the number of distinct fringe subtree-shapes in τ. We define $\mathcal{P}$ as the set of Pólya trees. This set of trees corresponds to the possible shapes of the recursive trees, once the increasing labelling has been removed. We denote by $\mathcal{P}_{\leq n}$ the set of all Pólya trees with size at most n. Then we have

$$\mathbb{E}\left(X_n\right) = \sum_{t \in \mathcal{P}_{\leq n}} \mathbb{P}(t \text{ occurs as subtree of } \tau) = \sum_{t \in \mathcal{P}_{\leq n}} 1 - \mathbb{P}(t \text{ does not occur as subtree of } \tau). \tag{6.2}$$

Recall that the tree t corresponds to a tree-shape, it is unlabelled, while τ is a recursive tree and therefore is increasingly labelled.

Now for a given Pólya tree $t \in \mathcal{P}$ let us consider a perturbed combinatorial class $\mathcal{S}_t$ that contains all recursive trees except for those that contain a t-shape as a (fringe) subtree. The corresponding exponential generating function satisfies the differential equation

$$S_t'(z) = \exp(S_t(z)) - P_t'(z), \tag{6.3}$$

where $P_t(z) = \ell(t)\frac{z^{|t|}}{|t|!}$, with $\ell(t)$ denoting the number of ways to increasingly label the tree-shape t.

So, using Equation (6.2) we obtain

$$\begin{aligned} \mathbb{E}\left(X_n\right) &= \sum_{t \in \mathcal{P}_{\leq n}} \left(1 - \mathbb{P}(t \text{ does not occur as shape of a fringe subtree of } \tau)\right) \\ &= \sum_{t \in \mathcal{P}_{\leq n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right). \end{aligned} \tag{6.4}$$

Therefore, the problem is now essentially reduced to the analysis of the asymptotic behaviour of $[z^n]S_t(z)$.

Solving Equation (6.3) we obtain the exponential generating function

$$S_t(z) = \ln\left(\frac{1}{1-\int_0^z \exp(-P_t(v))\,\mathrm{d}v}\right) - P_t(z). \tag{6.5}$$

Since $P_t(z)$ is not singular, the dominant singularity $\tilde{\rho}$ of $S_t(z)$ the following equation must hold:

$$\int_0^{\tilde{\rho}} \exp(-P_t(v))\,\mathrm{d}v = 1. \tag{6.6}$$

As $\exp(-P_t(v)) < 1$ for positive v, the dominant singularity $\tilde{\rho}$ is greater than 1. Therefore we write $\tilde{\rho} = \rho(1+\epsilon) = 1+\epsilon$ with suitable $\epsilon > 0$.

Notations

Before we proceed, let us introduce some frequently used notations: For the size and the weight of a Pólya tree t we use

$$k = |t| \quad \text{and} \quad w(t) = \frac{\ell(t)}{|t|!},$$

respectively. Moreover, let

$$G(z) := \int_0^z e^{-P_t(v)}\,\mathrm{d}v = \int_0^z e^{-w(t)v^k}\,\mathrm{d}v.$$

if $z \geq 0$ and its complex continuation if z is not a nonnegative real number. With this notation Equation (6.6) reads as $G(1+\epsilon) = 1$. By expanding the integrand, we obtain

$$G(z) = \sum_{\ell \geq 0} (-w(t))^\ell \frac{z^{\ell k+1}}{(\ell k+1)\cdot \ell!},$$

which shows that $G(z)$ is an entire function.

How to proceed

Taking a random recursive tree of size n, we are interested in the asymptotic behaviour of the size of the compacted tree issued from the compaction of the recursive one. In order to obtain bounds for this compacted size we proceed as follows: First, in Lemma 6.2.1, we compute a upper bound for $\tilde{\rho}$.

Then, in Lemma 6.2.4, we provide asymptotics for the n-th coefficient of the generating function $S_t(z)$ when n tends to infinity, thereby showing that the error term is uniform in the size k of the “forbidden” tree t.

The average size of a compacted tree corresponding to a random recursive tree is expressed as a sum over the forbidden trees. Thereby, the two cases, where the size k of the forbidden tree t is smaller or larger than $\log n$ are treated in a different way: Upper bounds for the size of the compacted tree are derived in Proposition 6.2.6 (small trees) and Proposition 6.2.7 (large trees). Finally, Proposition 6.2.9, gives a (crude) lower bound for the size of the compacted tree.

Lemma 6.2.1. *Let $S_t(z)$ be the generating function of the perturbed combinatorial class (*cf. *Equation* (6.3) *of recursive trees that do not contain a subtree of shape t and $\tilde{\rho}$ be the dominant singularity of $S_t(z)$ (*cf. *Equation* (6.6)*). Furthermore, let $k = |t|$ and $w(t) = \ell(t)/k!$ where $\ell(t)$ denotes the number of possible increasing labellings of the Pólya tree t. Then*

$$\tilde{\rho} = 1 + \epsilon < 1 + \frac{2w(t)}{k}.$$

Proof. First observe that the number of increasing labellings of the Pólya tree t is bounded by $(k-1)!$, which gives the very crude bound $w(t) \leq 1/k$.

Next, as $\tilde{\rho}$ satisfies $G(1+\epsilon) = 1$, it suffices to show the inequality $G\left(1 + \frac{2w(t)}{k}\right) > G(1+\epsilon)$. We show the equivalent inequality $G\left(1 + \frac{2w(t)}{k}\right) - G(1) > G(1+\epsilon) - G(1)$.

Then we have

$$G(1+\epsilon) - G(1) = 1 - \int_0^1 e^{-w(t)v^k}\,\mathrm{d}v \leq 1 - \int_0^1 (1 - w(t)v^k)\,\mathrm{d}v = \frac{w(t)}{k+1}.$$

On the other hand, if $k \geq 3$, then we have the lower bound

$$\begin{aligned} G\left(1 + \frac{2w(t)}{k}\right) - G(1) &\geq \frac{2w(t)}{k} \exp\left(-w(t)\left(1 + \frac{2w(t)}{k}\right)^k\right) \\ &\geq \frac{2w(t)}{k} \exp\left(-w(t)\left(1 + \frac{2}{k^2}\right)^k\right) \\ &= \frac{w(t)}{k} \cdot 2e^{-2w(t)} > \frac{w(t)}{k+1} \end{aligned}$$

which implies the assertion. In the course of this chain of inequalities we used $w(t) < 1/k$ and then $\left(1 + \frac{2}{k^2}\right)^k < 2$ (for $k \geq 3$) in the second line, then again $w(t) < 1/k$, and finally $k \geq 3$ and $2e^{-2/3} > 1$.

If $k = 2$, then t is a path of length one and therefore $w(t) = 1/2$. This gives explicitly $\int_1^{3/2} e^{-v^2/2}\,\mathrm{d}v > 1/6$ which is easily verified. □

Corollary 6.2.2. *With the notations of Lemma 6.2.1 we have the following asymptotic relation:*

$$\tilde{\rho} = 1 + \epsilon \sim 1 + \frac{w(t)}{k}, \text{ as } k \to \infty.$$

Proof. Write $G(z)$ as $G(z) = z + R(z)$ with

$$R(z) = \sum_{\ell \geq 1} (-w(t))^\ell \frac{z^{\ell k+1}}{(\ell k + 1) \cdot \ell!} \tag{6.7}$$

As $\tilde{\rho} = 1 + \epsilon$ is the smallest positive solution of $G(z) = 1$, it is the smallest positive zero of $z - 1 + R(z)$. From Lemma 6.2.1 we know that $\epsilon = \mathcal{O}\left(1/k^2\right)$ and thus $\tilde{\rho}^k \sim 1$, as k tends to infinity, and $R(\tilde{\rho}) = w(t)\tilde{\rho}^{k+1}/(k+1) + \mathcal{O}(1/k^3)$. This implies

$$\epsilon \sim \frac{w(t)}{k+1}\tilde{\rho}^{k+1} \sim \frac{w(t)}{k}, \tag{6.8}$$

as desired. □

Remark 6.2.3. *Using more terms of the expansion of $G(z)$, it is possible to derive a more accurate asymptotic expression for ϵ (in principle up to arbitrary order). As an example, we state*

$$\tilde{\rho} = 1 + \frac{w(t)}{k+1} + \frac{w(t)^2(3k+1)}{(k+1)(4k+2)} + \frac{w(t)^3(29k^3+32k^2+10k+1)}{6(k+1)^2(2k+1)(3k+1)} + \mathcal{O}\left(\frac{w(t)^4}{k}\right).$$

Now we are able to derive a uniform asymptotic expression for the coefficients of $S_t(z)$.

Lemma 6.2.4. *Let $S_t(z)$ be the generating function of the perturbed class of recursive trees seen in Equation (6.5). Then for sufficiently small $\delta > 0$ we have*

$$[z^n]S_t(z) = \frac{\tilde{\rho}^{-n}}{n}\left(1 + \mathcal{O}\left(\frac{1}{\sqrt{\ln n}}\right)\right), \text{ as } n \to \infty,$$

which holds uniformly for $D \le |t| \le n$, where $D > 0$ is independent of n and sufficiently large.

Proof. Recall that by Equation (6.5) we have

$$S_t(z) = \ln\left(\frac{1}{1-G(z)}\right) - P_t(z). \tag{6.9}$$

Since $G(z)$ is an entire function, the singularities of S_t are exactly the zeros of $G(z) - 1$. Therefore, consider z_0 such that $G(z_0) = 1$ and write $G(z) = z + R(z)$ with $R(z)$ as in Equation (6.7). Then

$$\begin{aligned} |R(z_0)| &\le \frac{1}{k+1}\sum_{\ell \ge 1} \frac{|w(t)|^\ell |z_0|^{k\ell+1}}{\ell!} \\ &< \frac{1}{k}(e^{|w(t)||z_0|^k} - 1) \end{aligned} \tag{6.10}$$

The first step is to show that $G(z) - 1$ does not have any zeros in a sufficiently large domain. We have to approach this in three steps, each enlarging the domain.

Assume first that $|z_0| \le 1 + \frac{e-1}{k}$. As the dominant singularity of $S_t(z)$ is $\tilde{\rho}$ and $\tilde{\rho} > 1$, we must have $|z_0| > 1$. Thus, the upper bound on $|z_0|$ and Equation (6.10) imply

$$1 - z_0 = R(z_0) = \mathcal{O}\left(1/k^2\right). \tag{6.11}$$

On the other hand, $R(z) \sim -\frac{w(t)}{k}z_0^k$ and $1 - z_0 \sim -w(t)/k$ because of Corollary 6.2.2. Thus z_0 is asymptotically equal to a k-th root of unity. But then $z_0 = \tilde{\rho}$, because the distance between the other k-th roots of unity and 1 is greater than $1/k$, which contradicts Equation (6.11).

Now assume that $|z_0| = 1 + \eta$ with $(e-1)/k < \eta < \ln(k)/k$. Then $w(t)|z_0|^k \le 1$ and so by Equation (6.10) we have then $|R(z_0)| \le (e-1)/k$. But we assumed $|z_0 - 1| > (e-1)/k$.

Finally, let $1 + \frac{\ln(k)}{k} < |z| \leq 1 + \frac{\ln k + \ln\ln\ln k}{k}$. In this region we have $|z-1| > \ln(k)/k$ but, using Equation (6.10), we get $|R(z)| \leq (\ln(k)-1)/k$ and thus $R(z)$ is too small to compensate the value of $z-1$. Indeed, we obtain that $|G(z)-1| > 1/k$.

Summarising what we have so far, we obtain that either $z_0 = \tilde{\rho}$ or $|z_0| > 1 + \frac{\ln k + \ln\ln\ln k}{k}$.

Notice that $G'(\tilde{\rho}) = \exp\left(-w(t)\tilde{\rho}^k\right) \neq 0$ and therefore $\tilde{\rho}$ is a simple zero of $G(z)-1$. Thus $G(z) - 1 = (z-\tilde{\rho})\tilde{G}(z)$ where $\tilde{G}(z)$ is analytic in the domain $|z| \leq 1 + \frac{\ln k + \ln\ln\ln k}{k}$ and does not have any zeros there. Thus,

$$\begin{aligned} S_t(z) &= \ln\left(\frac{1}{1-G(z)}\right) - P_t(z) \\ &= -\ln\left(1 - \frac{z}{\tilde{\rho}}\right) - \ln(\tilde{\rho}\,\tilde{G}(z)) - P_t(z), \end{aligned}$$

where, apart from the first summand, there are no singularities in $|z| \leq 1 + \frac{\ln k}{k}$. Expanding the logarithm gives

$$[z^n]S_t(z) = \frac{\tilde{\rho}^{-n}}{n}\left(1 + \mathcal{O}\left(n\tilde{\rho}^n[z^n]\ln\tilde{G}(z)\right)\right) \tag{6.12}$$

and we want to estimate $[z^n]\ln\tilde{G}(z)$ using Cauchy's estimate. Unfortunately, $\tilde{G}(z)$ is not uniformly bounded in k, so we have to analyse $\tilde{G}(z)$ a little more.

For applying Cauchy's estimate on the remainder function in Equation (6.12) we use the integration contour $|z| = 1 + \frac{\ln k + \ln\ln\ln k}{k}$. On this contour, we have

$$\frac{1}{k} \leq |G(z)-1| \leq |z-1| + |R(z)| \leq 3 + \frac{\ln k}{k}, \qquad \frac{\ln k}{k} < |z-\tilde{\rho}| < 3.$$

Since $\tilde{G}(z) = (G(z)-1)/(z-\tilde{\rho})$, this implies $|\ln\tilde{G}(z)| \leq \ln k + \ln 3$. Consequently, by Cauchy's estimate we get

$$[z^n]\ln\tilde{G}(z) = \mathcal{O}\left(\left(1 + \frac{\ln k + \ln\ln\ln k}{k}\right)^{-n}\ln k\right).$$

Finally, if k is sufficiently large, then

$$\begin{aligned} &\tilde{\rho}^n\left(1 + \frac{\ln k + \ln\ln\ln k}{k}\right)^{-n}\ln k \\ &\qquad\qquad \leq \left(1 + \frac{\ln k + \ln\ln\ln k}{k}\right)^{-n} \\ &\qquad\qquad \leq \left(1 + \frac{\ln n + \ln\ln\ln n}{n}\right)^{-n} \\ &\qquad\qquad = \mathcal{O}\left(\frac{1}{n\sqrt{\ln n}}\right), \end{aligned}$$

which yields the desired result after all. □

Remark 6.2.5. *Within this section many logarithms that occur are with respect to the base $\frac{1}{\sigma}$, where $\sigma \approx 0.338$ denotes the dominant singularity of the generating function of Pólya trees (*cf. [**FS09**, Section VII.5]*). To ensure a simpler reading we omit this base subsequently and just write* $\log n$ *instead. In order to distinguish, the natural logarithm will always be denoted by* $\ln n$*.*

Now we decompose the sum Equation (6.4) into

$$\mathbb{E}\left(X_n\right) = \sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k < \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) + \sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k \geq \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right), \tag{6.13}$$

and investigate the two sums individually, starting with the leftmost one, whose summands can be estimated by 1.

Proposition 6.2.6. *The first sum in (*Equation (6.13)*) behaves asymptotically as*

$$\sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k < \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n \to \infty}{=} \mathcal{O}\left(\frac{n}{\sqrt{(\log n)^3}}\right).$$

Proof. Remember that we have set $k := |t|$. Furthermore, we denote by $P(z)$ the generating function of Pólya trees and by σ its dominant singularity. Then

$$\begin{aligned}\sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k < \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) &\leq \sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k < \log n}} 1 \\ &= \sum_{k < \log n} [z^k]P(z) \sim \frac{1}{1-\sigma}[z^{\lfloor \log n \rfloor}]P(z) \\ &= \mathcal{O}\left(\frac{\sigma^{-\lfloor \log n \rfloor}}{\sqrt{(\log n)^3}}\right).\end{aligned}$$

Since $\log n$ has the base $1/\sigma$, we estimate $\sigma^{-\lfloor \ln n \rfloor} \leq n$, which completes the proof. □

Now we are able to estimate the asymptotic behaviour of the second sum in Equation (6.13).

Proposition 6.2.7. *Let $\mathcal{P}_{\leq n}$ denote the class of Pólya trees of size at most n. Then*

$$\sum_{\substack{t \in \mathcal{P}_{\leq n} \\ k \geq \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n \to \infty}{=} \mathcal{O}\left(\frac{n}{\log n}\right).$$

Proof. Using Lemma 6.2.4 we get, when n tends to infinity, that

$$\frac{[z^n]S_t(z)}{[z^n]T(z)} \underset{n \to \infty}{\sim} \tilde{\rho}^{-n} = (1+\epsilon)^{-n},$$

uniformly in $|t| = k$. Thus,

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{\sim} \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} \left(1-(1+\epsilon)^{-n}\right).$$

By means of the Bernoulli inequality we get

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} 1-(1+\epsilon)^{-n} \leq \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} n\cdot\epsilon,$$

which by use of Lemma 6.2.1 can be further simplified to

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} n\cdot\epsilon \underset{n\to\infty}{\sim} \sum_{k=\log n}^{n} \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}} n\cdot\frac{w(t)}{k} = \sum_{k=\log n}^{n} \frac{n}{k} \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}} w(t).$$

Using the fact that

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}} w(t) = [z^k]T(z) = \frac{1}{k},$$

we further get

$$\sum_{k=\log n}^{n} \frac{n}{k} \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}} w(t) = \sum_{k=\log n}^{n} \frac{n}{k^2} = \Theta\left(n\int_{\log n}^{\infty} \frac{1}{x^2}\,\mathrm{d}x\right) = \Theta\left(\frac{n}{\log n}\right).$$

Thus the statement is proved. □

Theorem 6.2.8. *Let X_n be the size of the compacted tree corresponding to a random recursive tree τ of size n. Then*

$$\mathbb{E}\left(X_n\right) \underset{n\to\infty}{=} \mathcal{O}\left(\frac{n}{\log n}\right).$$

Proof. The result follows directly by combining the previous propositions. □

Finally, we now prove a lower bound for the average size of the compacted tree based on a random recursive tree of size n.

Proposition 6.2.9. *Let $\mathcal{P}_{\leq n}$ denote the class of Pólya trees of size at most n. Then*

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{=} \Omega\left(\sqrt{n}\right).$$

Proof. First, we use the inequality $(1+\epsilon)^{-n} \leq \exp\left(-n\epsilon + \frac{n\epsilon^2}{2}\right)$ in order to estimate

$$\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) = \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq \log n}} \left(1-(1+\epsilon)^{-n}\right) \geq \sum_{k\geq \log n} \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}} \left(1 - e^{-n\epsilon+n\epsilon^2/2}\right). \tag{6.14}$$

For the sake of simplified reading we will use the abbreviation $\sum_t := \sum_{\substack{t\in\mathcal{P}_{\leq n}\\ |t|=k}}$ in the remainder of this proof. Since $x \mapsto 1-\exp\left(-nx+\frac{nx^2}{2}\right)$, $x \geq 0$, is a concave nonnegative function with a zero in the origin and $w(t) > 0$ for all t, we can estimate the inner sum in (Equation (6.14)), which yields

$$\sum_{k\geq\log n}\sum_t\left(1-e^{-n\epsilon+n\epsilon^2/2}\right) \geq \sum_{k\geq\log n}\left(1-\exp\left(-n\sum_t \epsilon+\frac{n}{2}(\sum_t \epsilon)^2\right)\right).$$

Note that ϵ depends on t, and that

$$\sum_t \epsilon \underset{n\to\infty}{\sim} \sum_t \frac{w(t)}{k} = \frac{1}{k}\sum_t w(t) = \frac{1}{k^2}.$$

Thus, we get

$$\begin{aligned}\sum_{\substack{t\in\mathcal{P}_{\leq n}\\ k\geq\log n}}\left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right) &\geq \sum_{k\geq\log n}\left(1-\exp\left(-\frac{n}{k^2}+\frac{n}{2k^4}\right)\right)\\ &\underset{n\to\infty}{\sim}\int_{\ln n}^{\infty}\left(1-\exp\left(-\frac{n}{x^2}+\frac{n}{2x^4}\right)\right)\,\mathrm{d}x\\ &=\sqrt{n}\int_{\sqrt{n}\log n}^{\infty}\left(1-\exp\left(-\frac{1}{y^2}+\frac{1}{2ny^4}\right)\right)\,\mathrm{d}y.\end{aligned}$$

Since the integral is convergent this gives a lower bound that is $\Theta(\sqrt{n})$. □

6.3 Average compression of plane binary trees under increasingly labelled distribution

Plane binary increasing trees have a classical specification in the context of Analytic Combinatorics, once again by using the Greene operator, or boxed product, allowing to define increasing labelling constraint for decomposable objects. Thus the specification of this class $\mathcal{T}$ is

$$\mathcal{T} = \mathcal{Z}^{\square}\star(1+\mathcal{T})^2. \tag{6.15}$$

This specification defines a tree to be rooted with an atom $\mathcal{Z}$ associated to a pair of elements that are either the empty element (representing no subtree) or a subtree itself from the class $\mathcal{T}$. Once again the operator $\cdot\ ^{\square}\star\ \cdot$ ensures the fact that the smallest available label must be used for the atom $\mathcal{Z}$.

In Figure 6.6 we have represented on the left-hand side a plane increasing binary tree structure containing 5.000 nodes. It has been uniformly sampled among all trees with the same size. The original root of the tree is represented using a small circle $\circ$. On the right-hand side, we have depicted the nodes that are kept after the compaction of the latter tree. It remains only 1,361 nodes.

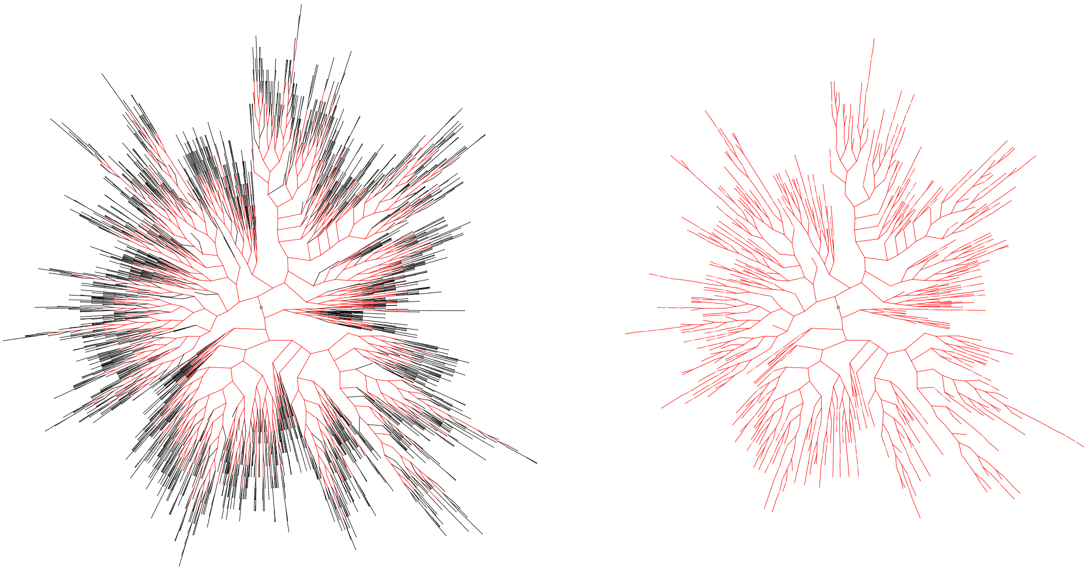


Figure 6.6: (left) a uniformly sampled (plane) increasing binary tree of size 5,000: black fringe subtrees are removed by the compaction; (right) the red head is of size 1,361

By using the symbolic method [FS09], the latter specification Equation (6.15) translates as

$$T(z) = \int_0^z (1+T(v))^2 \, \mathrm{d}v,$$

in terms of $T(z)$ the exponential generating function for $\mathcal{T}$. We can also rewrite it as a differential equation

$$T'(z) = (1+T(z))^2, \qquad \text{with } T(0) = 0$$

The equation can be solved such that

$$T(z) = \frac{z}{1-z},$$

with the dominant singularity $\rho = 1$.

The exponential generating function $S_t(z)$ of the perturbed class of plane increasing binary trees that do not contain the tree-shape t (where t is a non-labelled binary tree) as a fringe subtree, fulfils the equation

$$S_t'(z) = (1+S_t(z))^2 - P_t'(z) \qquad \text{with } S_t(0) = 0 \tag{6.16}$$

where $P_t(z) = \frac{\ell(t) z^{|t|}}{|t|!}$ and $\ell(t)$ denotes the number of ways to increasingly label the plane binary tree t. The quantity $\ell(t)$ is also called the hook length of t and it is well known that $\ell(t)$ equals $|t|!$ divided by the product of the sizes of all fringe subtrees of t (*cf.* e.g. [Knu98, p.67] or [BGP16]). We first start with a lemma establishing an upper bound for the normalised hook length.

Lemma 6.3.1. *Let t be a binary tree of size k. By defining the weight of the tree t as $w(t) := \frac{\ell(t)}{k!}$, where $\ell(t)$ denotes the hook length of t, we have*

$$w(t) \leq \frac{1}{2^{k-2}}.$$

Proof. Recall that the hook length equals $|t|!$ divided by the product of the sizes of all fringe subtrees s of t. If we write $s \leq t$ to say that s is a fringe subtree of t, then this means that $w(t) = 1/\prod_{s\,:\,s\leq t} |s|$. Consider now a tree t. If $k = 1$, then t is a single node and hence $w(t) = 1$. Otherwise, the root of t has children being roots of fringe subtrees. If $s \leq t$, then either $s = t$ and so $|s| = k$ or s is one of the fringe subtrees of one of the subtrees rooted at a child of the root of t. Therefore

$$w(t) = \begin{cases} \frac{1}{k} w(t') & \text{if the root of } t \text{ has one child } t' \\ \frac{1}{k} w(t_\ell) w(t_r) & \text{if the root of } t \text{ has the two children } t_\ell \text{ and } t_r. \end{cases}$$

Now proceed by induction: Set $w_n := max_{t\,:\,|t|=n} w(t)$. Then we have obviously that $w_n = \max\{w_\ell \cdot w_{n-1-\ell} \mid \ell = 0..n-1\}/n$ with $w_0 = 1$. For the first seven values a direct computation shows

$$(w_1, w_2, \dots, w_7) = \left(1, \frac{1}{2}, \frac{1}{3}, \frac{1}{8}, \frac{1}{15}, \frac{1}{36}, \text{and } \frac{1}{63}\right).$$

As the first seven values of the sequence $1/2^{k-2}$ are

$$2, 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \text{and } \frac{1}{32},$$

we assume that the result is correct until $k-1$.

Let t be a binary tree of size k. If the root of t has only one child t' of size $k-1$, then by induction we obtain

$$w(t) = \frac{w(t')}{k} \leq \frac{1}{k\ 2^{k-3}} \leq \frac{1}{2^{k-2}}.$$

Otherwise, the root of t has two children. Let us denote the corresponding fringe subtrees by t_ℓ of size ℓ and t_r of size $k-\ell-1$, (with $\ell < k$). By the induction hypothesis, we have $w(t_\ell) \leq 1/2^{\ell-2}$ and $w(t_r) \leq 1/2^{k-\ell-3}$ and thus

$$w(t) = \frac{1}{k} w(t_\ell) w(t_r) \leq \frac{1}{k}\ \frac{1}{2^{k-5}} = \frac{8}{k}\ \frac{1}{2^{k-2}},$$

which is smaller than $1/2^{k-2}$ for $k \geq 8$. □

Finally, note that the inverse term by term of our sequence corresponds to the sequence stored as EIS A132862.

By the same combinatorial argument as in the previous section we know that $S_t(z)$ has a unique dominant singularity $\tilde{\rho}$, which is greater than the dominant singularity $\rho = 1$ of $T(z)$. Thus, we set again $\tilde{\rho} = \rho(1+\epsilon) = 1+\epsilon$. Since Equation (6.16) is a Riccati differential equation (*cf.* [**Inc44**] for a background on Riccati equations), we use the ansatz $S_t(z) = \frac{-u'(z)}{u(z)}$ to get the transformed equation

$$u''(z) - 2\, u'(z) + (1 - w(t)kz^{k-1})\, u(z) = 0, \tag{6.17}$$

where we use the same abbreviations as in the previous section, namely $k := |t|$ and $w(t) := \frac{\ell(t)}{k!}$. Note that the condition $S_t(0) = 0$ implies $u'(0) = 0$ and $u(0) \neq 0$.

The singularities of a function $u(z)$ solving a linear differential equation (with polynomial coefficients) are given by the singularities of the coefficient of the highest derivative, *i.e.*, in our case the coefficient of $u''(z)$, which is 1. The reader can refer to Miller [**Mil06**] for details. Thus, we can conclude that $u(z)$ is an entire function. As a direct consequence we know that the singularities of $S_t(z)$ are given by the zeros of $u(z)$ (that are not zeros of $u'(z)$) and are therefore poles. More precisely the dominant singularity $\tilde{\rho}$ must be a simple pole for $S_t(z)$, since for $u(z) = (\tilde{\rho} - z)^l v(z)$, (such that ρ is not a zero of $v(z)$), it follows that $u'(z) = -(\tilde{\rho} - z)^{l-1} v(z) + (\tilde{\rho} - z)^l v'(z)$. Thus

$$S_t(z) = \frac{l}{\tilde{\rho} - z} - \frac{v'(z)}{v(z)},$$

which implies

$$S_t(z) \underset{z \to \tilde{\rho}}{\sim} \frac{l/\tilde{\rho}}{1 - z/\tilde{\rho}}.$$

Taking the derivative we get $S_t'(z) \sim \frac{1}{\tilde{\rho}^2} \frac{l}{(1 - z/\tilde{\rho})^2}$. Plugging in the asymptotic expressions for S_t and S_t' in the original differential Equation (6.16) we get

$$\frac{1}{\tilde{\rho}^2} \frac{l}{\left(1 - \frac{z}{\tilde{\rho}}\right)^2} \underset{z \to \tilde{\rho}}{\sim} \left(1 + \frac{l/\tilde{\rho}}{1 - \frac{z}{\tilde{\rho}}}\right)^2,$$

since the monomial P_t is analytic in $\tilde{\rho}$. Comparing the main coefficients yields $l = 1$, and thus $\tilde{\rho}$ is a simple zero of the function $u(z)$ and

$$S_t(z) \underset{z \to \tilde{\rho}}{\sim} \frac{1}{\tilde{\rho} - z}.$$

How to proceed

As in the previous section, we have a singularity $\tilde{\rho} = 1 + \epsilon$ with $\epsilon > 0$ depending on t, or k. In order to get results on the average size of the compacted tree of a random increasing binary tree we proceed similarly to the recursive tree case. Lemma 6.3.3 gives an asymptotic expression for $\tilde{\rho}$ that quantifies its dependence on t, when the size k of the "forbidden" tree tends to infinity.

As a next step, Lemma 6.3.4 shows that $S_t(z)$ has a unique dominant singularity $\tilde{\rho}$ on the circle of convergence, which is used in Lemma 6.3.5 to obtain the asymptotic behaviour of the coefficients of the generating function $S_t(z)$.

Again, the average size of a compacted tree can be represented as a sum over the forbidden trees, where we distinguish between the two cases whether the size of the trees is smaller or larger than $\log n$ in order to get an upper bound (see Proposition 6.3.7 and Proposition 6.3.8). This time the lower bound is obtained in Theorem 6.3.10 by carefully analysing the sum bounds and which uses estimate for the weights $w(t)$ (see Lemma 6.3.1). The final Θ result is then obtained as an immediate corollary.

We start from the equation $u''(z) - 2\ u'(z) + (1 - w(t)kz^{k-1})\ u(z) = 0$ with the initial conditions $u(0) = \gamma$, and $u'(0) = 0$. The value γ can be chosen arbitrarily, as $S_t(z) = \gamma u'(z)/(\gamma u(z))$, and thus, γ cancels. For simplification reasons in the following we choose $u(0) = -1$ together with the initial condition $u'(0) = 0$.

Lemma 6.3.2. *The function $u(z)$ defined by the differential Equation (6.17) and the initial conditions $u(0) = -1$ and $u'(0) = 0$ satisfies*

$$\begin{aligned} u(z) =& z e^z \sum_{m\geq 0} \left(\frac{w(t)k}{(k+1)^2}\right)^m \frac{1}{m!\ (m+\alpha)_m} z^{(k+1)m} \\ &- e^z \sum_{m\geq 0} \left(\frac{w(t)k}{(k+1)^2}\right)^m \frac{1}{m!\ (m-\alpha)_m} z^{(k+1)m}, \end{aligned}$$

where $(x)_m$ denotes the falling factorials $(x)_m = x(x-1)\cdots(x-m+1)$ and $\alpha = 1/(k+1)$.

Before starting with the proof, note our computer algebra system suggests a solution of Equation (6.17) as a linear combination of Bessel functions. Before proving the latter statement, let us recall the context of Bessel functions. The reader can refer to the book of Bender and Orszag [**BO99**] for more details. The ordinary differential equation

$$z^2\ y''(z) + z\ y'(z) + (z^2 - \alpha^2)\ y(z) = 0,$$

with α not being an integer is such that the solutions $y(z)$ are linear combination of the Bessel functions $J_\alpha(z)$ and $Y_\alpha(z)$ defined as

$$\begin{aligned} J_\alpha(z) &= \sum_{n=0}^{\infty} \frac{(-1)^n}{n!\ \Gamma(n+\alpha+1)} \left(\frac{z}{2}\right)^{2n+\alpha} \qquad \text{and} \\ Y_\alpha(z) &= \frac{J_\alpha(z)\cos(\alpha\pi) - J_{-\alpha}(z)}{\sin(\alpha\pi)}. \end{aligned}$$

Proof. In order to be closer to the Bessel equation, we define a new function,

$$y(z) := u(z) \cdot \exp(-z)/\sqrt{z},$$

thus we get an new equation for $y(z)$:

$$y''(z) + \frac{1}{z}\ y'(z) - \left(\frac{1}{4z^2} + w(t)kz^{k-1}\right)\ y(z) = 0,$$

with $\lim_{z\to 0+} y(z) = -\infty$ and $\lim_{z\to 0+} y'(z) = +\infty$. Let us now introduce the following change of variable $x := \left(\frac{k+1}{2\sqrt{-w(t)k}} z\right)^{2/(k+1)}$. After simplification we obtain

$$\beta^2\ y''(\beta) + \beta\ y'(\beta) + \left(\beta^2 - \frac{1}{(k+1)^2}\right)\ y(\beta) = 0,$$

with $\beta = \frac{2\sqrt{-w(t)k}}{k+1} t^{(k+1)/2}$. We recognise the Bessel equation and thus $y(\beta)$ is a linear combination of the functions $J_\alpha(\beta)$ and $Y_\alpha(\beta)$.

A first remark is necessary while reading the expression for $u(z) = \sqrt{z}\exp(z)y(z)$. At a first sight, it seems that the solution is not analytic at 0 due to the factor $\sqrt{z}$. But this is only an artefact in the way we present $u(z)$ through a linear combination of Bessel functions. We recall that using Equation (6.17) we previously proved that $u(z)$ is an entire function.

Let us now introduce the following functions

$$f(z) = \sqrt{z}\exp(z)J_\alpha\left(2\tilde{\beta}z^{\frac{1}{2\alpha}}\right)$$

and

$$\bar{f}(z) = \sqrt{z}\exp(z)J_{-\alpha}\left(2\tilde{\beta}z^{\frac{1}{2\alpha}}\right),$$

with $\tilde{\beta} := \frac{\sqrt{-w(t)k}}{k+1}$ and $\alpha := \frac{1}{k+1}$. Due to the relationship between the function $u(z), y(\beta)$ and the Bessel functions, we deduce $u(z)$ is a linear combination of the functions $f(z)$ and $\bar{f}(z)$. Let us write first it as $u(z) = \lambda f(z) + \bar{\lambda}\bar{f}(z)$ and now let us find both constants λ and $\bar{\lambda}$. Using the series expression for $J_.(\cdot)$ we notice both functions $f(z)$ and $\bar{f}(z)$ are analytic and can be expanded around 0 as

$$f(z) = \tilde{\beta}^\alpha\frac{1}{\Gamma(1+\alpha)}z + \dots \qquad \text{and} \qquad \bar{f}(z) = \tilde{\beta}^{-\alpha}\frac{1}{\Gamma(1-\alpha)} + \tilde{\beta}^{-\alpha}\frac{1}{\Gamma(1-\alpha)}z + \dots.$$

Thus we deduce

$$u(0) = -1 = \bar{\lambda}\tilde{\beta}^{-\alpha}\frac{1}{\Gamma(1-\alpha)}, \qquad \text{and} \qquad u'(0) = 0 = \frac{\lambda\tilde{\beta}^\alpha}{\Gamma(1+\alpha)} + \frac{\bar{\lambda}\tilde{\beta}^{-\alpha}}{\Gamma(1-\alpha)}.$$

By using $\frac{\Gamma(1+\alpha)}{\Gamma(m+1+\alpha)} = \frac{1}{(m+\alpha)_m}$, where $(x)_m$ denotes the falling factorials $(x)_m = x(x-1)\dots(x-m+1)$, we conclude

$$\begin{aligned} u(z) =& ze^z\sum_{m\geq 0}\left(\frac{w(t)k}{(k+1)^2}\right)^m\frac{1}{m!\,(m+\alpha)_m}z^{(k+1)m} \\ &- e^z\sum_{m\geq 0}\left(\frac{w(t)k}{(k+1)^2}\right)^m\frac{1}{m!\,(m-\alpha)_m}z^{(k+1)m}. \end{aligned}$$

□

We are now ready to analyse the dominant singularity of $S_t(z)$.

Lemma 6.3.3. *Let $S_t(z)$ be the generating function of the perturbed combinatorial class of plane increasing binary trees that do not contain the shape t as a subtree (of size k). With $\tilde{\rho}$ denoting the dominant singularity of $S_t(z)$, we get*

$$\tilde{\rho} = 1 + \epsilon \underset{k\to\infty}{\sim} 1 + \frac{2w(t)}{k^2},$$

where $w(t) = \frac{\ell(t)}{k!}$ and $\ell(t)$ denotes the hook length of t.

Proof. For combinatorial reasons we deduced that the equation $u(z) = 0$ must have a solution $\tilde{\rho} > 1$ and no smaller positive solution. When k tends to infinity we expect that $\tilde{\rho} = 1 + \epsilon$ tends to 1, *i.e.* ϵ tends to 0.

First observe that $u(0) = -1$ and

$$u\left(1+\frac{1}{k^2}\right) = \frac{1}{k^2} + \mathcal{O}\left(\frac{w(t)}{k}\right) > 0,$$

as $w(t)$ decays exponentially due to Lemma 6.3.1. Thus $\epsilon = \mathcal{O}\left(1/k^2\right)$ and plugging $z = 1+\epsilon$ into $u(z) = 0$ gives then

$$\epsilon + (1+\epsilon)^{k+1}\frac{w(t)k}{(k+1)^2}\left(\frac{1+\epsilon}{1+\alpha} - \frac{1}{1-\alpha}\right) = \mathcal{O}\left(\frac{w(t)^2}{k^2}\right).$$

This implies $\epsilon - 2w(t)/k^2 = \mathcal{O}\left(w(t)^2/k^2\right)$ and hence $\epsilon \sim 2w(t)/k^2$, which finishes the proof. □

So, Lemma 6.3.3 ensures that for $|t| = k$ tending to infinity the generating function $S_t(z)$ has a dominant singularity at $\tilde{\rho} \sim 1 + 2w(t)/k^2$. Now we show that in a circle with radius smaller than $1 + 2\ln(k)/k$ there is no other singularity for $S_t(z)$.

Lemma 6.3.4. *Let $\tilde{\rho}$ be the dominant singularity of $S_t(z)$. Then, for all $\delta > 0$ the following assertion holds: If k is sufficiently large, then the generating function $S_t(z)$ does not have any singularity in the domain $\tilde{\rho} < |z| < 1 + \frac{(2-\delta)\ln k}{k}$.*

Proof. First let us remember that the singularities of $S_t(z)$ are given by the zeros of the function $u(z)$ that is defined in Lemma 6.3.2. Now let us write $\tilde{u}(z) := u(z)\exp(-z)$ and note that $u(z)$ and $\tilde{u}(z)$ have the same zeros. Thus, in the remainder of this proof we investigate $\tilde{u}(z)$, which can be written as $\tilde{u}(z) = zF(z) - G(z)$ with

$$F(z) = \sum_{m\geq 0}\left(\frac{w(t)k}{(k+1)^2}\right)^m \frac{1}{m!}\frac{1}{(m+\alpha)_m} z^{(k+1)m}, \quad \text{and}$$

$$G(z) = \sum_{m\geq 0}\left(\frac{w(t)k}{(k+1)^2}\right)^m \frac{1}{m!}\frac{1}{(m-\alpha)_m} z^{(k+1)m},$$

still with $\alpha := 1/(k+1)$. Therefore we get

$$\begin{aligned}|F(z) - G(z)| &= \left|\sum_{m\geq 0}\left(\frac{w(t)k}{(k+1)^2}\right)^m \frac{1}{m!}\left(\frac{1}{(m-\alpha)_m} - \frac{1}{(m+\alpha)_m}\right) z^{(k+1)m}\right| \\ &= \mathcal{O}\left(\frac{w(t)}{k}\,\alpha\,|z|^{k+1}\right) = \mathcal{O}\left(\frac{w(t)}{k^2}|z|^{k+1}\right).\end{aligned}$$

Now, let us rewrite $\tilde{u}(z)$ as

$$\tilde{u}(z) = (z-1)F(z) + F(z) - G(z), \tag{6.18}$$

set $|z| = 1 + \eta$ and perform a distinction of two cases:

- $\eta = \mathcal{O}\left(1/k\right)$: This implies $|z|^{k+1} = \Theta(1)$ for k tending to infinity. Thus $F(z) \sim 1$, $G(z) \sim 1$, and then $F(z) - G(z)$ tends to 0 when k tends to infinity. Furthermore, Equation (6.18) implies $\tilde{u}(z) \sim z - 1$. The equation $\tilde{u}(z) = 0$ therefore yields $z - 1 \sim F(z) - G(z)$, which is $\mathcal{O}\left(w(t)/k^2\right)$. Since we know that $\tilde{\rho} \sim 1 + \frac{2w(t)}{k^2}$ we get $|z-1| = \Theta(\tilde{\rho} - 1)$.

But for zeros z_0 of $\tilde{u}(z)$ with $|z_0| = 1 + o\left(1/k\right)$ we know $z_0 - 1 \sim (2w(t)/k^2) \cdot z_0^k \sim 2w(t)/k^2$, which is equivalent to $z_0^k \sim 1$. Hence $z_0 \sim \sqrt[k]{1} = \cos\left(\frac{2\pi}{k}\right) + i\sin\left(\frac{2\pi}{k}\right)$ and

$$\tilde{\rho}\sqrt[k]{1} \sim \left(1 + \frac{2w(t)}{k^2}\right)\left(1 - \frac{2\pi^2}{k^2} + i\frac{2\pi}{k}\right) \sim 1 + i\frac{2\pi}{k},$$

which is a contradiction to $z_0 - 1 \sim 2w(t)/k^2$. Thus, the function $\tilde{u}(z)$ has no zeros for $\tilde{\rho} < |z| \leq 1 + \mathcal{O}\left(1/k\right)$.

- $\eta = C_k/k$, with $C_k \leq (2-\delta)\ln k$, and C_k tends to infinity with k: In this case we have $|z|^{k+1} \sim e^{C_k} = o(k^2)$, and thus $|F(z) - G(z)| = o(w(t))$ and $F \sim 1 + o\left(w(t)k\right) \sim 1$ when k tends to infinity. Using again Equation (6.18) yields $\tilde{u}(z) = z - 1 + o(w(t)) \sim z - 1$. Since $|z| = 1 + \eta$ we have $|z - 1| \geq C_k/k$ and because of $o(w(t)) = o\left(1/k\right)$ we know that $\tilde{u}(z)$ cannot be zero in $\tilde{\rho} < |z| < 1 + ((2-\delta)\ln k)/k$.

□

Now we are interested in the ratio $[z^n]S_t(z)/[z^n]T(z)$, which corresponds to the probability that a random plane binary tree of size n does not contain the binary tree shape t as a fringe subtree.

Lemma 6.3.5. *Let $T(z)$ be the generating function of plane increasing trees and $S_t(z)$ the generating function of the perturbed class that has the dominant singularity $\tilde{\rho}$. Then, for any $\eta > 0$ we have*

$$\frac{[z^n]S_t(z)}{[z^n]T(z)} \underset{n\to\infty}{=} \tilde{\rho}^{-n-1}\left(1 + \mathcal{O}\left(\frac{\ln n}{n^{1-\eta}}\right)\right),$$

uniformly for $D \leq k \leq n$, if D is sufficiently large (but independent of n).

Proof. First, let us remember that $\tilde{\rho}$ is a unique zero of the function $u(z)$. Thus, we can write

$$u(z) = \left(1 - \frac{z}{\tilde{\rho}}\right)v(z), \tag{6.19}$$

with $v(\tilde{\rho}) \neq 0$ and by Lemma 6.3.4 we additionally know that $v(z) \neq 0$ in $\tilde{\rho} < |z| < 1 + \frac{(2-\delta)\ln k}{k}$, provided that k is sufficiently large. Furthermore, we have

$$u'(z) = \left(1 - \frac{z}{\tilde{\rho}}\right)v'(z) - \frac{1}{\tilde{\rho}}\,v(z),$$

which yields

$$S_t(z) = \frac{1}{\tilde{\rho} - z} - \frac{v'(z)}{v(z)}.$$

Thus,

$$[z^n]S_t(z) = \tilde{\rho}^{-n-1} - [z^n]\frac{v'(z)}{v(z)} = \tilde{\rho}^{-n-1} - (n+1)[z^{n+1}]\ln v(z). \tag{6.20}$$

Now, we estimate the second summand in Equation (6.20). First we use a Cauchy coefficient integral to write

$$n[z^n]\ln v(z) = \frac{n}{2\pi i}\int_{\mathcal{C}} \frac{\ln v(t)}{t^{n+1}}\,\mathrm{d}t, \tag{6.21}$$

where the curve $\mathcal{C}$ is described by $|t| = 1 + \frac{(2-\delta)\ln k}{k}$ with some $\delta > 0$. The absolute value of the logarithm of $v(z)$ is given by $|\ln v(z)| = \left|\ln\left(|v(z)|e^{i\arg v(z)}\right)\right| = |\ln|v(z)| + i\arg(v(z))|$. Furthermore, by Equation (6.19) we have $|v(z)| = |u(z)|/\,|1 - z/\tilde{\rho}|$, which can be estimated along $\mathcal{C}$ via

$$|v(z)| \leq \frac{|u(z)|k}{(2-\delta)\ln k}.$$

Now, we have to estimate $|u(z)|$. By Lemma 6.3.2 we get

$$|u(z)| \leq \sum_{m\geq 0}\left(\frac{w(t)}{k}\right)^m \frac{1}{m!}\left|\frac{z}{(m+\alpha)_m} - \frac{1}{(m-\alpha)_m}\right| |z|^{(k+1)m}.$$

Along $\mathcal{C}$ we have $|z|^{(k+1)m} \leq (k^{2-\delta})^m$ and the absolute value $\left|\frac{z}{(m+\alpha)_m} - \frac{1}{(m-\alpha)_m}\right|$ can be estimated by $\left|\frac{z}{(m+\alpha)_m} - \frac{1}{(m-\alpha)_m}\right| \leq \frac{2+\mu}{(m-\alpha)_m}$, for some $\mu > 0$ which results in

$$|u(z)| \leq \sum_{m\geq 0}(w(t)k^{1-\delta})^m \frac{2+\mu}{m!(m-\alpha)_m} \leq K,$$

for a constant K independent of k.

Putting all together, we can estimate the integral Equation (6.21) by

$$\begin{aligned} n[z^n]\ln v(z) &= \frac{n}{2\pi i}\int_{\mathcal{C}} \frac{\ln v(t)}{t^{n+1}}\,\mathrm{d}t \\ &\leq n(\ln k + \ln K - \ln((2-\delta)\ln k)\left(1 + \frac{(2-\delta)\ln k}{k}\right)^{-n-1} \\ &\leq n\ln n\left(1 + \frac{(2-\delta)\ln k}{k}\right)^{-n} \end{aligned}$$

which implies the following asymptotic relation:

$$[z^n]S_t(z) = \tilde{\rho}^{-n-1}\left(1 + \mathcal{O}\left(n\ln n\left(1 + \frac{(2-\delta)\ln k}{k}\right)^{-n}\tilde{\rho}^n\right)\right)$$

Finally, note that for sufficiently large k we have the estimate

$$\begin{aligned} \tilde{\rho}\left(1 + \frac{(2-\delta)\ln k}{k}\right)^{-1} &\leq \left(1 + \frac{(2-2\delta)\ln k}{k}\right)^{-1} \\ &\leq \left(1 + \frac{(2-2\delta)\ln n}{n}\right)^{-1} \end{aligned}$$

and, as

$$\left(1 + \frac{(2-2\delta)\ln n}{n}\right)^{-n} = \mathcal{O}\left(n^{-2+2\delta}\right),$$

we obtain the assertion by setting $\eta = 2\delta$. □

Now, we separate the sum of interest, *i.e.* $\sum_{t\in\mathcal{B}} \mathbb{P}\left[t \text{ occurs at subtree of } \tau\right]$, where τ denotes a plane increasing binary tree of size n and $\mathcal{B}$ denotes the class of (unlabelled) plane binary trees, analogously as we did in the previous section for recursive trees.

Remark 6.3.6. *Now our underlying class of tree-shapes is the class of plane binary trees and no more the class of instead of Pólya trees. Since the dominant singularity of the generating function of binary trees is* $1/4$*, we use henceforth* $\log n$ *as an abbreviation for the logarithm with respect to base* 4.

$$\mathbb{E}\left(X_n\right) = \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k<\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) + \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right). \tag{6.22}$$

In order to estimate the first sum, we proceed analogously to Proposition 6.2.6.

Proposition 6.3.7. *Let* $B(z)$ *be the generating function associated to* $\mathcal{B}$*, of (unlabelled) binary trees, whose dominant singularity is* $1/4$*. Then asymptotically when* n *tends to infinity we have*

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k<\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{=} \mathcal{O}\left(\frac{n}{\sqrt{(\log n)^3}}\right).$$

Proof. A crude estimate gives

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k<\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \leq \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k<\log n}} 1 = \sum_{k<\log n} [z^k]B(z) \underset{n\to\infty}{\sim} \frac{1}{1-\frac{1}{4}}[z^{\lfloor \log n\rfloor}]B(z)$$
$$\underset{n\to\infty}{=} \mathcal{O}\left(\frac{\left(\frac{1}{4}\right)^{-\lfloor \log n\rfloor}}{\sqrt{(\log n)^3}}\right).$$

This is already sufficient, since $\log n = \log_4 n$ and thus $\left(\frac{1}{4}\right)^{-\lfloor \log n\rfloor} \leq n$, which completes the proof. □

Estimating the second sum in (Equation (6.22)) works analogously to the proof of Proposition 6.2.7 in the previous section.

Proposition 6.3.8. *Let* $\mathcal{B}_{\leq n}$ *denote the class of binary trees of size at most* n*. Then*

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{=} \mathcal{O}\left(\frac{n}{\log n}\right).$$

Proof. Using Lemma 6.3.5 we get that for n tending to infinity

$$\frac{[z^n]S_t(z)}{[z^n]T(z)} \underset{n\to\infty}{\sim} \tilde{\rho}^{-n-1} = (1+\epsilon)^{-n-1}$$

Thus,

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{\sim} \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1-(1+\epsilon)^{-n-1}\right).$$

Bernoulli's inequality then gives

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} 1-(1+\epsilon)^{-n-1} \leq \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} (n+1)\cdot\epsilon,$$

which by use of Lemma 6.3.3 further simplifies to

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} (n+1)\cdot\epsilon \underset{n\to\infty}{\sim} \sum_{k=\log n}^{n} \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ |t|=k}} (n+1)\cdot\frac{2w(t)}{k^2} = \sum_{k=\log n}^{n} \frac{2n}{k^2} \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ |t|=k}} w(t).$$

But since the inner sum equals 1, we finally get

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} (n+1)\cdot\epsilon \underset{n\to\infty}{=} \sum_{k=\log n}^{n} \frac{2n}{k^2} \underset{n\to\infty}{=} \Theta\left(n\int_{\log n}^{\infty}\frac{1}{x^2}\,\mathrm{d}x\right) \underset{n\to\infty}{=} \Theta\left(\frac{n}{\log n}\right). \qquad \square$$

Theorem 6.3.9. *Let X_n be the size of the compacted tree corresponding to a random binary tree of size n. Then*

$$\mathbb{E}(X_n) \underset{n\to\infty}{=} \mathcal{O}\left(\frac{n}{\log n}\right).$$

Proof. The result follows directly by combining the previous propositions. □

Theorem 6.3.10. *Let X_n be the size of the compacted tree corresponding to a random binary tree of size n. Then*

$$\mathbb{E}(X_n) \underset{n\to\infty}{=} \Omega\left(\frac{n}{\log n}\right).$$

Proof.

$$\begin{aligned}\mathbb{E}(X_n) &= \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k<\log n}} \left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right) + \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right)\\ &\geq \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq\log n}} \left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right)\\ &\geq \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} \left(1-\frac{[z^n]S_t(z)}{[z^n]T(z)}\right)\end{aligned}$$

Using Lemma 6.3.5 we get that for n tending to infinity

$$\frac{[z^n]S_t(z)}{[z^n]T(z)} \underset{n\to\infty}{\sim} \tilde{\rho}^{-n-1} = (1+\epsilon)^{-n-1}$$

Thus,

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{\sim} \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} \left(1-(1+\epsilon)^{-n-1}\right).$$

We have that from Lemma 6.3.1

$$w(t) = \frac{w(t')}{k} \leq \frac{1}{k\ 2^{k-3}} \leq \frac{1}{2^{k-2}}.$$

And we have from Lemma 6.3.3

$$\epsilon \underset{k\to\infty}{\sim} \frac{2w(t)}{k^2}.$$

Therefore:

$$\epsilon \underset{k\to\infty}{=} \mathcal{O}\left(\frac{2}{2^{k-2}k^2}\right),$$

and when $2\ \log n \leq k \leq n$,

$$-(n-1)\epsilon \underset{k\to\infty}{=} \mathcal{O}\left(\frac{1}{k^2}\right).$$

And thus,

$$(1+\epsilon)^{-n-1} \underset{n\to\infty}{\sim} 1-(n-1)\epsilon.$$

Finally,

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} 1-(1+\epsilon)^{-n-1} \sim \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} (n+1)\cdot\epsilon,$$

The rest of the calculations are the same as in the last two lines of the proof of Proposition 6.3.8, since the lower bound of the sum only changes by a constant which does not change the Θ result.

If we sum up what we have obtained

$$\mathbb{E}\left(X_n\right) \geq \sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right),$$

and

$$\sum_{\substack{t\in\mathcal{B}_{\leq n}\\ k\geq 2\log n}} \left(1 - \frac{[z^n]S_t(z)}{[z^n]T(z)}\right) \underset{n\to\infty}{=} \Theta\left(\frac{n}{\log n}\right).$$

And we have

$$\mathbb{E}\left(X_n\right) = \Omega\left(\frac{n}{\log n}\right),$$

as desired. □

Corollary 6.3.11.

$$\mathbb{E}\left(X_n\right) \underset{n\to\infty}{=} \Theta\left(\frac{n}{\log n}\right).$$

Proof. The proof is a direct consequence of Theorem 6.3.9 and Theorem 6.3.10. □

6.4 A compressed data structure

The probability model induced by plane increasing binary trees is the classical permutation model of *binary search trees* (or bst). Thus the typical shape of a uniformly sampled plane increasing binary tree consisting of n internal nodes corresponds to the typical shape of a binary search tree built using a uniform random permutation of n elements. See Drmota [**Drm09**, Section 1.3.3] for details about the latter correspondence. Thus the tree structure of a typical bst has the properties we have found out in the previous section. In particular, by removing the information stored in the nodes the typical compaction of the tree gives a compacted structure consisting of $\Theta(n/\ln n)$ nodes (on average).

Our objective here is to desing a new lossless immutable data structure based on the tree structure induced by the compaction of a bst to which we associate some extra information in the nodes and in the edges to keep all the information and keep the same bounds to searching. We also make experiments related to this data structure in the end of this section.

The main idea is ta have an efficient database for searching elements while this databse occupy the least possible memory space. However, we compressing the data structure it becomes immutable and therefore only searching operations can be done on it. We can access but not modify it efficiently. Of course since the compaction is lossless, it is always possible to decompress it, modify it as we want and then compress it once again.

The bst built for example on the permutation $(4, 8, 6, 2, 9, 1, 3, 7, 5)$ is represented with the classical tree structure in the left-hand side of Figure 6.7. This example will be used as an illustration throughout the whole section. In order to compress the tree structure, first the node labels must be removed, as presented before. Thus by using a compaction through a postorder traversal of the tree, the example becomes the tree structure presented in the

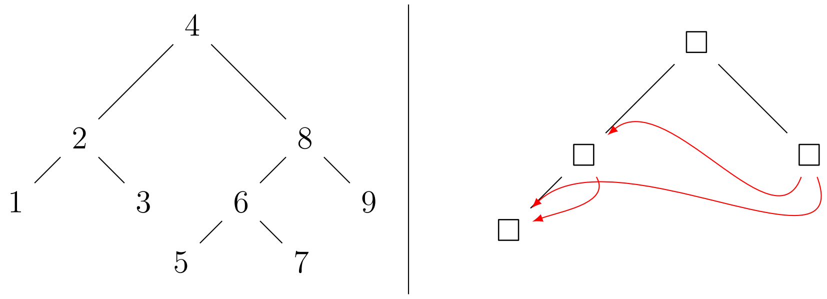


Figure 6.7: (left) A bst built e.g. on $(4, 8, 6, 2, 9, 1, 3, 7, 5)$; (right) The compacted tree structure associated to the bst

right-hand side Figure 6.7. By adding the values stored in the original bst we get the tree of Figure 6.8. When a substructure has been removed through the compaction process, then in addition to the red pointer, the list of the labels, obtained through a *preorder traversal* of the substructure is stored. The latter, associated to the size of the substructures, depicted with the circled blue values, allows to obtain an efficient research. Let us present an example. We would like to know if 7 is stored in the structure. 7 is larger than 4, thus from the root we take the right edge to reach 8. The value we are looking for is smaller than 8. We take the left red pointer, and take also in consideration the list $L := [6, 5, 7]$. We define an index $i = 0$ corresponding to the actual index in the list we are interested in. Using the pointer, we reach 2 that corresponds in fact to $L[0] = 6$. Since 7 is larger than 6, we must follow the right child of 2, thus the new index is $i := i + 2$ (the list stores the values obtained through the preorder traversal), the constant 2 is the size of the left subtree attached to 2 plus 1 for the node labeled by 2. Now $L[2] = 7$, we have reached the value we were interested in.

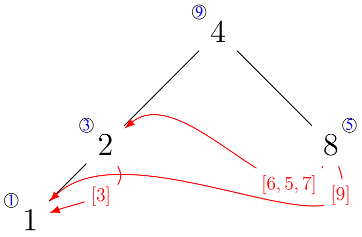


Figure 6.8: Labeled compacted structure associated to the original bst

Proposition 6.4.1. *In the compacted* bst *containing n values, the search complexity is the same as in the* bst *with respect to the number of value comparisons. There may be an extra-cost corresponding to the number of additions (related to the index) to traverse a list. The number of additions is at most equal to the number of comparisons to search for the value.*

Proof. The number of value comparisons is exactly the same in the compacted structure as in the original bst. In fact, we just share the identical unlabeled tree structure, thus the number of comparisons does not change. For the same reason, if we must search inside a list associated to a red edge, then, for each comparison there is one addition to shift inside the list. □

In the following Figure 6.9 we have represented two experiments through our python prototype. In the left-hand side we are interested in the compaction ratio between the compressed data structure and the original bst. Here we are interested in the whole size needed in memory. In particular the size of the integer values is counted but further the data structure size itself is important. It is this latter that is in fact compressed: in the bst many pointers are needed to reach the nodes of the tree. Many pointers and nodes are replaced in the compressed data by lists of integers that need much less memory in practice. In the figure, in the abscissa we represent the number of integers stored in the data structures; and in the ordinate, we compute the ratio between the size in memory of the compressed data structure in front of the size of its corresponding bst. Each dot corresponds to one sample, and the green curve is the average value among all samples. The experiments are starting with 250 integer values up to 20,000 with steps every 250 values, and for each size we have used 30 uniformly sampled bsts. We observe that even for small bsts, the compression ratio is very interesting, smaller than 0.5. Further we remark that the green curve looks like the theoretical result: it is very close to a function $x \mapsto \alpha / \ln x$ for a given α.

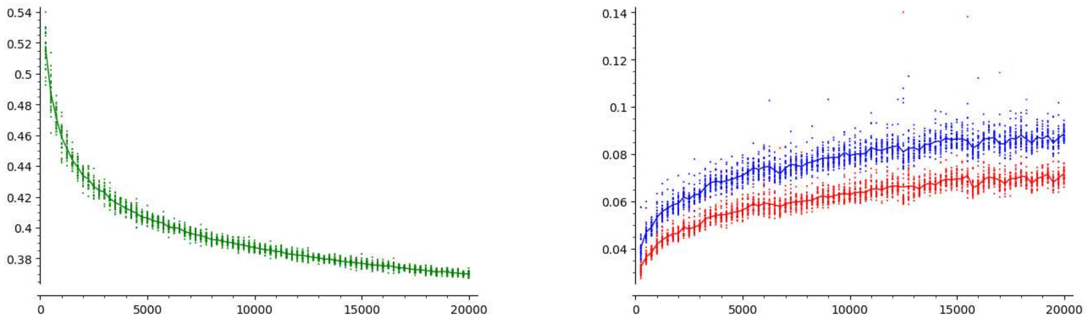


Figure 6.9: (left) Experimental compression ratio; (right) Experimental search time comparison

In the right-hand side of Figure 6.9, for the same set of bsts and associated compacted structures, we search for 1,000 randomly sampled values present in the two structures. Each red dot is the average time, in milliseconds, (among the 1,000 searches) for finding the value inside the bst, and the blue point is the analogous time for the search in the compressed structure. For both complexity measures (number of comparisons or of arithmetic additions) the average complexity stays of the same order $O(\ln n)$ as for the original bst, as we see it in the figure. By computing the ratio of the blue values and the red values, the mean seems oscillating around 1.25 for the whole range of sampled structures.

Let us conclude this section with the following remark. The point of view we have chosen is to build first the bst and then, once the insertion and deletion process is done, we convert the bst into a compressed data structure that is used only for search. We could develop a prototype data structure that manages insertion in deletion but the efficiency would probably be much less than the one of bst, because of the substructure recognition problem.

6.5 Conclusion

For the case of *binary trees* under *increasingly labelled distribution* (see Definition 6.1.1) we were able to get a Θ result, by showing that if we have random tree of size n, denoted X_n, the size of the compacted tree on average is

$$\mathbb{E}(X_n) = \Theta\left(\frac{n}{\ln n}\right).$$

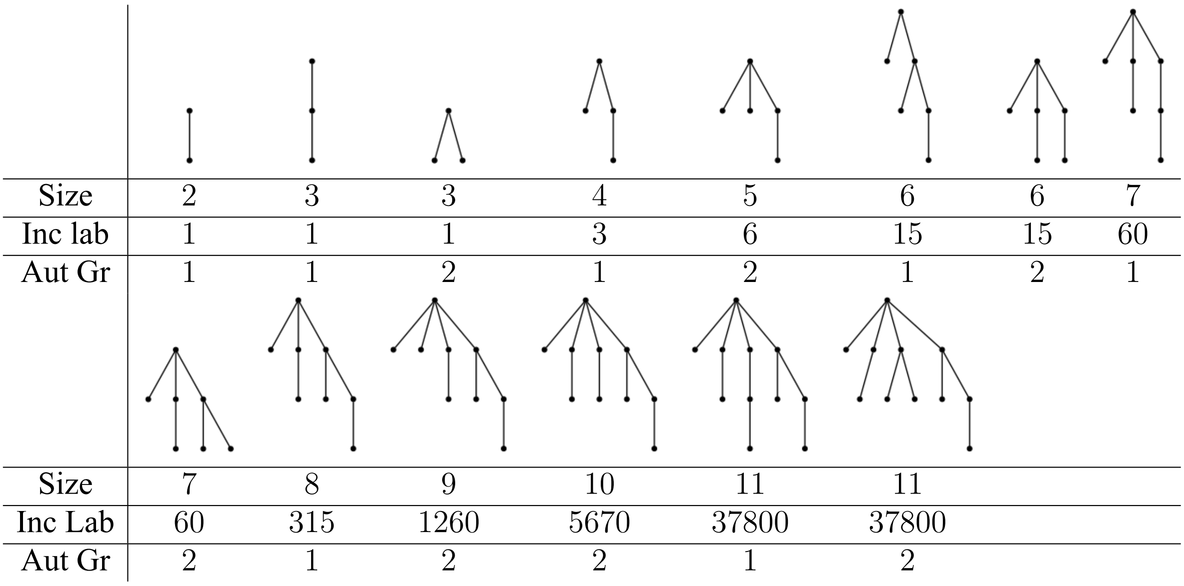
However, for *Pólya trees* under *increasingly labelled distribution* the size of the compacted tree is smaller asymptotically than the average size of *Pólya trees* under *uniform distribution*.

More precisely, we proved that if we have a random tree of size n, denoted Y_n, the compacted tree is on average of size

$$C_1\sqrt{n} \leq \mathbb{E}(Y_n) \leq C_2\,\frac{n}{\ln n}.$$

For some constants C_1 and C_2.

Numerical simulations suggest that this upper bound is already sharp, *i.e.*, that the size of the compacted tree is $\Theta\left(\frac{n}{\ln n}\right)$. However, in order to prove this conjecture, one has to find the distribution of the weights $w(t)$, which is very challenging task for non-plane trees due to the appearance of automorphisms. In Figure 6.10 we have depicted *Pólya trees* that give the maximum number of labellings for the first sizes. We also gave their number of increasing labellings as well as the size of their automorphism group.



Size	2	3	3	4	5	6	6	7
Inc lab	1	1	1	3	6	15	15	60
Aut Gr	1	1	2	1	2	1	2	1

Size	7	8	9	10	11	11
Inc Lab	60	315	1260	5670	37800	37800
Aut Gr	2	1	2	2	1	2

Figure 6.10: Pólya trees that give the maximum number of increasing labellings for $2 \leq n \leq 11$. 'Inc Lab' stands for the number of increasing labellings of the tree and 'Aut' Gr the size of the automorphism group of the tree.

Open question (Maximum number of labellings of non-plane trees). Let $\mathcal{P}_n$, be the set of all Pólya trees of size n and let $\ell(t)$ denotes the number of ways to increasingly label a tree t. We define the sequence M_n as follows

$$M_n = max(\{l(t) \mid t \in \mathcal{P}_n\}).$$

The sequence M_n represents the number of labellings of the tree in $\mathcal{P}_n$ that gives the maximum of number increasing labellings. How does the sequence M_n behaves and how does its asymptotics looks like?

Thus, obtaining the (maximum) number of labellings of non-plane trees of a given size is still work in progress, with the aim to improve the lower bounds such that we can show the Θ-result.

Open question (Average compaction rate for an any class of trees under increasingly labelled distribution). Based on our theoretical and numerical results we conjecture the following.

Conjecture 6.5.1. *Let $\mathcal{T}$ be a rooted class of trees. Let X_n be a random tree of size n generated according to the increasingly labelled distribution defined in Definition 6.1.1. Then the average size of the compacted version of X_n is*

$$\mathbb{E}(X_n) = \Theta\left(\frac{n}{\ln n}\right).$$

We explain the choice of the two classes of increasing trees, that were investigated within this paper. The reason to choose recursive trees and increasing binary trees was that for these two classes our computer algebra system is able to solve the differential equation defining $S_t(z)$, although in case of increasing binary trees the solution is already more complicated and involves some Bessel functions. However, in case of the third prominent class of increasing trees, ports (plane oriented recursive trees), we did not get any explicit solution for the analogous of $S_t(z)$; thus this case is still an open question.

As a final note, remember the way we have compacted the bsts in the last section. Using a pointer to describe the erased fringe subtree and the list of the labels in a specific traversal (labels that must be kept in the compacted tree), we are able to search in the compacted structure efficiently. But more generally, the way we have compacted the tree can be used for all possible tree structures. In the original paper [**FSS90**] by Flajolet *et al.*, the authors compact only identical fringe subtrees in simply generated trees. We focus on the tree structure and its compaction as well, but the probability model on the tree shapes is a different one, induced by the labelling. Moreover, we use a different additional information management in order to cope with labels and could there extend the compaction to labelled tree models. It is desirable to study other natural labelled tree classes and the resulting compaction ratio.

CHAPTER 7

Random generation

Contents

> La création a toujours besoin de hasard.[1]
>
> Jacques Godbout (1933 -), Les Têtes à Papineau

[1] Creation always needs chance

7.1 Introduction

Boltzmann sampling is a method for constructing random generation algorithms for combinatorial structures. *Boltzmann sampling* is closely related to the *Boltzmann distribution* used in statistical mechanics. The idea of applying Boltzmann samplers for combinatorial classes has been introduced in 2004 by Duchon, Flajolet, Louchard and Schaeffer in [**DFLS04**]. The idea is to get automatic random generation algorithms directly from the combinatorial specification (symbolic method). A later paper by Flajolet, Fusy and Pivoteau extended the admissible operators in [**FFP07**]. However, these papers did not include the operators for increasingly labelled structures such as increasing trees. Bodini, Roussel and Soria showed in [**BRS12**] how to extend the samplers to include specifications that give rise to differential equations of the first order as it is the case for in *increasing trees*. The thesis of Bodini [**Bod10**] and Roussel [**Rou12**] are dedicated to the study of *Boltzmann samplers*. The thesis of Dien [**Die17**] has also a dedicated chapter for *Boltzmann samplers* on increasing classes of graphs.

Boltzmann sampling has the great advantage of being very fast in time complexity, they can also be tuned to generate structures with some fixed parameters (multivariate combinatorial classes) in [**BP10**] and [**BBD18**].

However, *Boltzmann samplers* are less efficient for exact size sampling. Since they are based on an expected size n for the generated object. There exists some methods in order to get exact size generation with a rejection principle. But then the time complexity, in most cases the complexity becomes quadratic instead of the linear complexity for the approximate size sampling.

On the other hand the ***recursive generation method*** makes use of the recursive specification of the combinatorial class in order to get systematic uniform random generation algorithms. Many of the basic principles of this approach have been developed in the 70's by Nijenhuis and Wilf [**NW75**]. This approach has the been adapted to the analytic combinatorics point of view in [**FZV94**] by Flajolet, Van Custem and Zimmerman.

Unranking algorithms form also part of this global approach. The idea is to first define a total order over the objects under consideration so that to each object corresponds exactly one integer. Then, after choosing an integer it is possible to build deterministically the associated object. Moreover, if the rank is uniformly chosen among all possible ranks, then the unranking algorithm becomes a uniform random sampler. The unranking approach gives also a way for obtaining an exhaustive sampler, just by iterating the sampling over all possible ranks. In [**BDGV18**], the authors give an example of both methods: recursive generation and unranking method. Other accounts can be found in the books of Ruskey [**Rus03**] and the one by Alonso and Schott [**AS95**].

The *recursive generation* method generally operates in two steps. First, it necessitates a step of pre-computations of some information such that the enumeration sequence of the combinatorial class. The more information is stored in the beginning, the more efficient will be the second step. The second step is for sampling random objects. The pre-computations are done once before the sampling, then any number of objects can be sampled.

This chapter is organised as follows. Our main objective is to sample any variety of trees generated by the *evolution process* presented in Chapter 5. As we will see a uniform random generation algorithm will not have a polynomial complexity in the general case. However, for the three classes presented in Chapter 4, there exist fast algorithms to efficiently sample these trees uniformly at random with exact size generation. We present them in Section 7.2 under the sections Section 7.2.1, Section 7.2.2 and Section 7.2.3

Then in Section 7.3 we focus our attention on the general case and give an unranking algorithm to generate *uniformly* a tree of a fixed size in the specified variety. This algorithm necessitates the generating of all *integer partitions* of size n, which does not have a polynomial complexity.

7.2 Efficient uniform samplers for the three models of increasing Schröder trees

In the next three sections we present the *recursive generation algorithms* for the three classes of trees presented in Chapter 4. For these three models there are simplifactions in the recurrence that allow us to have efficient uniform samplers which is not the case in general.

In Figure 7.1 we compare between the complexities of the uniform samplers for each model of increasing Schröder trees that are presented in Algorithm 1, Algorithm 3 and Algorithm 5.

7.2.1 Increasing Schröder

In this section, we present an algorithm that samples a Schröder tree uniformly at random among all Schröder trees of a given size. Our aim is to use this algorithm to generate trees of large size (typically several thousands of leaves): we thus provide a detailed analysis of the complexity of our sampler.

Note that the uniform sampling of structures with increasing labelling constraints is not so classical in the context of analytic combinatorics. Martínez and Molinero [**MM03, Mol05**] focus on the recursive method: using and generalising recursive and unranking generation methods, they give a method that, given a combinatorial specification, automatically outputs a uniform generation algorithm and its complexity analysis. Using a different approach based on Boltzmann generation, Bodini, Roussel and Soria [**BRS12**] give an algorithmic framework to develop Boltzmann samplers in the context of specifications that lead to differential equation of the first order.

The paper [**BDF$^+$16**] show that this framework can be extended to the context of differential equations of higher order; in particular, they apply this method to the generation of diamonds satisfying differential equations of order 2.

The bijection presented in Section 4.2 immediately gives an algorithm that samples a tree uniformly among all Schröder trees of size n: first sample a permutation uniformly at random among all permutations of size n in $\mathcal{HP}$, and then build its image by $\mathcal{M}$. While there exists

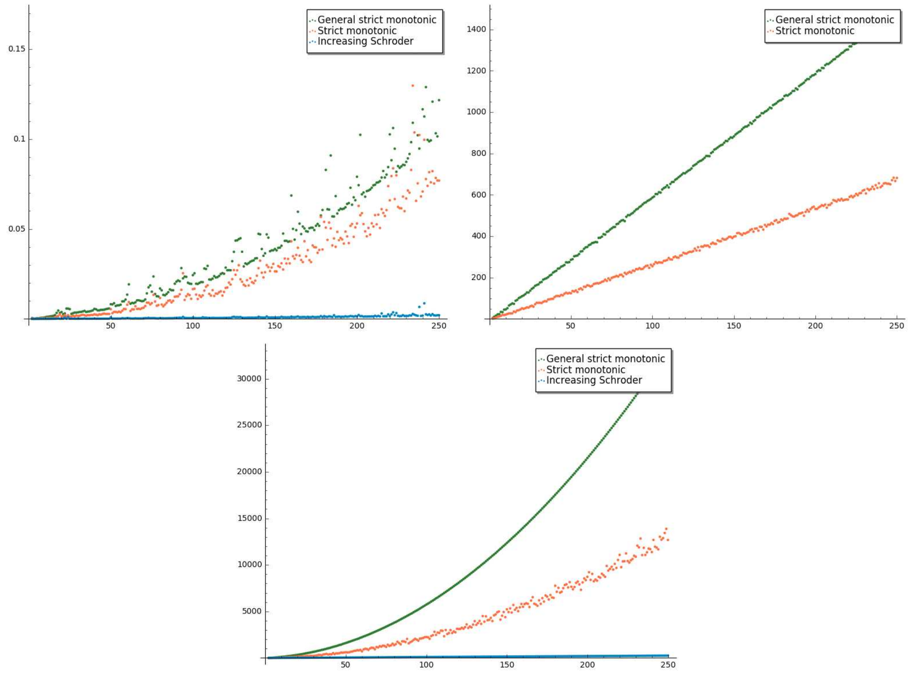


Figure 7.1: Complexity of uniform samplers for the the three models of increasing Schröder trees. (up left) Time complexity of the sampling in milliseconds. (up right) Arithmetic operations on big numbers complexity. (down) Arithmetic operations complexity.

fast algorithms to sample permutations (see for example [**BBHT17**]), it is not clear how to make the application of $\mathcal{M}$ efficient.

Instead, we use the bijection $\mathcal{M}$ as a basis for a direct *probabilistic construction*. Indeed, one can sample a uniform bijection uniformly at random in $\mathcal{H}P$ by doing the following recursive procedure: if $n = 2$, then return $\sigma^{(2)} = (2, 1)$. If $n \geq 3$, assume we have sampled $\sigma^{(n-1)}$ uniformly among all permutations of size $n-1$. Draw an integer k_n uniformly at random in $\{1, \dots, n\}$, and set $\sigma^{(n)}_n = k_n$, and

$$\sigma_i^{(n)} = \begin{cases} \sigma_i^{(n-1)} & \text{if } \sigma_i^{(n-1)} < k_n \\ \sigma_i^{(n-1)} + 1 & \text{otherwise.} \end{cases}$$

One can indeed check that $\sigma^{(n)}$ is uniformly distributed among all permutations of size n in $\mathcal{HP}$. Executing this random sampling of $\sigma^{(n)}$ simultaneously with $\mathcal{M}$ (note that, for all $n \geq 3$, $\sigma^{(n-1)} = \hat{\sigma}^{(n)}$, where the notation $\hat{\sigma}$ is defined in the definition of $\mathcal{M}$) is the idea of our sampler:

Using the adequate data structures, as for example by keeping an array of pointers to all leaves and another one to the last inserted internal node, each insertion in the tree under construction is done in constant time. We thus get

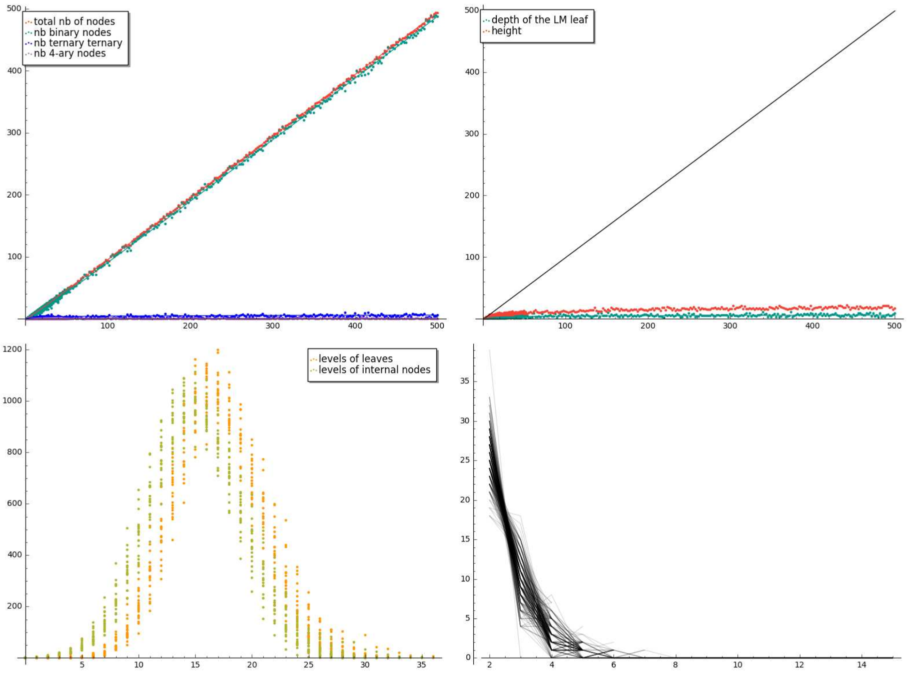


Figure 7.2: Simulation of different parameters on *Increasing Schröder trees*. (left) The number of d-ary nodes. (right) The height and the depth of the (LM) leftmost leaf. The sizes of the sampled trees range from 1 to 500. (down left) The profile of trees of size 10000. (down right) Root degree, the lines gets darker as n moves from 2 to 250, there are 40 trees sampled for each size

Algorithm 1 Increasing Schröder Tree Builder

```
function TreeBuilder($n$)
    if $n = 1$ then
        return the single leaf
    $T$ = the root labelled by 1 and attached to two leaves
    $\ell = 2$
    for $i$ from 3 to $n$ do
        $k = rand\_int(1, i)$
        if $k = i$ then
            Add a new leaf to the last added internal node in $T$
        else
            Create a new binary node at position $k$ in $T$
                with label $\ell$ and attached to two leaves
            $\ell = \ell + 1$
    return T
```

The function $rand\_int(a, b)$ returns uniformly at random an integer in $\{a, a+1, \ldots, b\}$.

Theorem 7.2.1. *The function* TreeBuilder(n) *in Algorithm Algorithm 1 is a uniform sampling algorithm for size n trees. Asymptotically, it operates in $\mathcal{O}(n)$ operations on trees and necessitates $\mathcal{O}(n \ln n)$ random bits.*

Since the generation times is almost linear it is possible to generate trees of large sizes, up to 10000 in few seconds. In Figure 7.2 we simulate these trees with some parameters. Some of these parameters have already been studied in Section 4.2, namely the average number of d-ary nodes in the tree, the depth of the leftmost leaf and the average height of the tree. However, the profile of the tree has not been studied yet.

7.2.2 Strict monotonic Schröder

To sample uniformly at random a strict monotonic Schröder tree of size n, we could choose a two-step algorithm. First we sample uniformly an ordered partition of the set $\{1, \ldots, n-1\}$ and then with the use the bijection of Section 4.3 we transform it into a strict monotonic Schröder tree. But here, in this section, we prefer to present a direct algorithm that generates uniformly a strict monotonic Schröder tree, i.e. without the intermediate step of generating another combinatorial object like an ordered partition.

We will use the *recursive generation method* which will turn out to give as directly an unranking algorithm as a by-product of the process.

For both types of algorithms (unranking or recursive generation) some pre-computations are done (only once before the sampling of many objects). We compute (and store) the numbers of trees of sizes from 1 to n. This calculation is be done with a quadratic complexity (in the number of arithmetic operations) using the recursive formula for $(g_n)_{n\geq 1}$ (see Equation (4.22)). This complexity is only achieved if we first compute and memorise all values of $(i!)_{1\leq i\leq n}$. Then it only remains to build the tree of rank r recursively. If r is sampled uniformly at random in $\{0, 1, \ldots, g_n - 1\}$ the algorithm is a uniform sampler and if r is deterministically chosen, then the algorithm is a classical unranking algorithm. To do this, we recall that (see Equation (4.18)), for all $n \geq 1$,

$$g_n = \binom{n-1}{n-2} g_{n-1} + \binom{n-1}{n-3} g_{n-2} + \cdots + \binom{n-1}{0} g_1, \tag{7.1}$$

and interpret this equation combinatorially: to build a tree of size n, we take a size $\ell \in \{1, \ldots, n-1\}$ tree T_ℓ constructed with exactly one less iteration. To grow it into a size-n tree, we interpret the binomial coefficient $\binom{n-1}{\ell-1}$ as the number of composition of n in ℓ parts: some of the ℓ leaves of T_ℓ are replaced by some internal nodes to which leaves are attached, some leaves remain leaves. To do that we traverse the tree T_ℓ and each time we see a leaf, we do the following action: if the next part (in the composition) is of value 1, we keep the leaf unchanged otherwise for a value $s > 1$, we replace the leaf by an internal node (well labelled with the currently step number) and attached s leaves to it. We then take the next part of the composition into consideration and continue the tree traversal.

Focusing on Equation (7.1) and the equation above we see that a function allowing the unranking of compositions is necessary. Recall the composition of the integer n into ℓ parts is in bijection with the number of combinations of $(\ell-1)$ elements chosen in $(n-1)$ ones. A way to prove it consists in laying $(\ell-1)$ barriers in the sequence of n bullets in order to define ℓ parts. There are classical algorithms to unrank combinations in the lexicographical order. A first algorithm has been described by Buckles and Lybanon [**BL77**]. Another, more efficient, has just been settled in the technical report [**DGH**]. For both of them we can easily prove

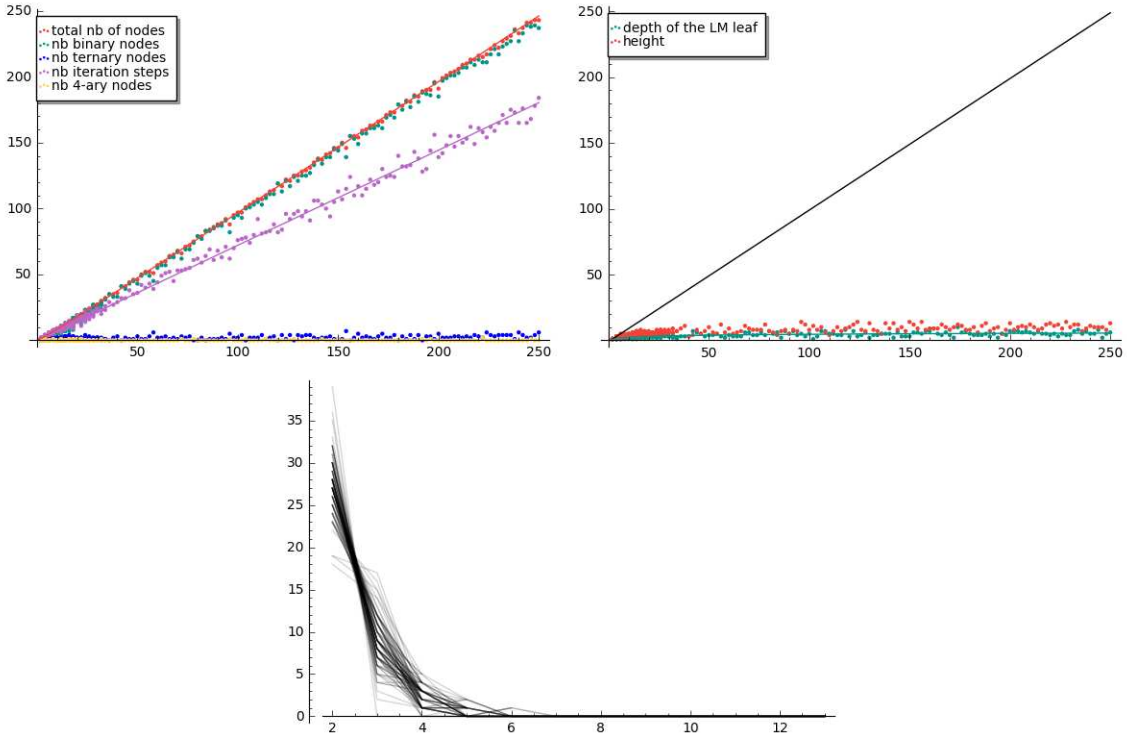


Figure 7.3: Simulation of different parameters on *Strict monotonic Schröder trees*. (left) The number of d-ary nodes and the number of iteration steps. (right) The height and the depth of the (LM) leftmost leaf. The sizes of the sampled trees range from 1 to 250. (down) Root degree, the lines gets darker as n moves from 2 to 100, with 40 trees sampled for each size.

that their average complexity (when ℓ rages over all possibilities) is $\Theta(n)$ in the number of arithmetic operations by having first memoized all factorial values of the numbers from 0 to n. In the following we develop a simpler approach based on the classical recursive generation without any lexicographic constraint like in the two mentioned papers. The algorithm is an unranking method for the composition of integers. It is based on the reverse lexicographic order (cf e.g. [**Rus03**]) so that we get an easier implementation[2]. For simplification, we suppose having memoized all values of $\binom{r}{s}$ for $r \in \{1, n\}$ and $s \in \{1, r\}$. Using the classical Pascal's rule for binomial coefficients, we obtain the following recurrence for the number of composition of n into ℓ:

$$C_{n,\ell} = \binom{n-1}{\ell-1} = C_{n-1,\ell} + C_{n-1,\ell-1}. \tag{7.2}$$

We thus deduce Algorithm Algorithm 2 for the unranking method.

[2] For the composition unranking, note that it would suffice to look for the rank $\binom{n}{\ell} - 1 - r$ (instead of r) in order to get the lexicographic order.

Algorithm 2 Reverse Lexicographic Composition Unranking

```
function UnrankComposition($n, \ell, r$)
    if $n = \ell$ and $r = 0$ then
        return $(1, 1, \ldots, 1)$
    if $r < \binom{n-2}{\ell-1}$ then
        $C$ :=UnrankComposition($n - 1, \ell, r$)
        $C[0] := C[0] + 1$
        return $C$
    else
        $s := r - \binom{n-2}{\ell-1}$
        $C := (1) \cup$ UnrankComposition($n - 1, \ell - 1, s$)
        return $C$
```

Theorem 7.2.2. *The function* UnrankComposition *is an unranking algorithm (based the reverse lexicographic order) and calling it with the parameters $\ell \leq n$ and a uniformly-sampled integer r in $\{0, \ldots, \binom{n-1}{\ell-1} - 1\}$, gives as output a uniform composition of n in ℓ parts.*

Using the memorisation of binomial coefficients, the algorithm needs at most $(\ell - 1)$ arithmetic operations on big integers.

Proof. We prove that the algorithm is correct by induction on n. The result is true when $n = \ell = r = 1$ since the algorithm returns (1). Fix an integer n and assume that the algorithm is correct for all $\ell \leq n-1$, and that the total order over compositions is the reverse lexicographic one (see, e.g., [**Rus03**] for the definition of the reverse lexicographic order). Let ℓ be an integer between 0 and n, and r be an integer chosen uniformly at random in $\{0, \ldots, C_{n,\ell} - 1\}$. Equation Equation (7.2) implies that a composition of n in ℓ parts is either a composition of $(n - 1)$ in ℓ parts whose first part has been increased by one, or it is a composition of $(n - 1)$ in $(\ell - 1)$ parts, and a new part equal to 1 is added at the beginning of the composition. In both cases, the first elements are all greater than the second elements according to the lexicographic order. The recurrence hypothesis ends the proof since the rank value r (or s in the second case) is adapted to each of the latter cases.

The number of arithmetic operations is direct when all binomial coefficients are first memorised. □

In Equation (7.1) the first term is much bigger than the second one, which is much bigger than the third one and so on. This approach, focusing first on the dominant terms is an adaptation to the idea underlying the *Boustrophedonic order* presented in [**FZV94**]. It allows to improve essentially the average complexity of the random sampling algorithm. In our case of strict monotonic Schröder trees do not follow a standard specification (cf. [**FZV94**] for details), the complexity gain is even better. The loop starting in line 6 aims at determining the interesting term in the sum Equation (7.1), thus the size of the tree in the evolution process letting to build the tree of rank s and size n.

The traversal $\mathcal{T}$ used to substitute some leaves in line 13 determines partly the total order over the strict monotonic trees. Let α be an strict monotonic tree, and $\mathcal{T}$ a given traversal of all trees. Remark that there is a single evolution process building α (the construction is

Algorithm 3 Strict monotonic Schröder Tree Unranking

```
 1: function UnrankTree($n, s$)
 2:     if $n = 1$ then
 3:         return the tree reduced to a single leaf
 4:     $\ell := 1$
 5:     $r := s$
 6:     while $r >= 0$ do
 7:         $r := r - \binom{n-1}{\ell} \cdot g_{n-\ell}$
 8:         $\ell := \ell + 1$
 9:     $\ell := \ell - 1$
10:     $r := r + \binom{n-1}{\ell} \cdot g_{n-\ell}$
11:     $T :=$UnrankTree($n - \ell, r \mod g_{n-\ell}$)
12:     $C :=$UnrankComposition($n, n - \ell, r // g_{n-\ell}$)
13:     Substitute in $T$, using traversal $\mathcal{T}$, some leaves according to $C$
14:     return the tree $T$
```

The sequences $(g_\ell)_{\ell \leq n}$ and $(\ell!)_{\ell \in \{1,\ldots,n\}}$ have been pre-computed and stored.
Line 13: The operation $//$ is the Euclidean division.

unambiguous). If α is built at the step ℓ, then we denote by $\tilde{\alpha}$ the single tree (built with $\ell - 1$ steps) and $\underline{\alpha}$ the single composition such that at step ℓ replacing the leaves from $\tilde{\alpha}$ according to the composition $\underline{\alpha}$, using the traversal $\mathcal{T}$, we obtain α.
Here we remark that the whole tree α is strongly dependent from the traversal of the leaves of $\tilde{\alpha}$ (while some leaves are substituted by an internal nodes attached to new leaves according to $\underline{\alpha}$). We define now how to compare strict monotonic trees (we use the analogous notations than the latter for all trees).

Definition 7.2.3. *Let α and β be two trees. We define $\alpha < \beta$ if*

- *the size of α is smaller than the one of β, or*
- *if both sizes are equal to n and if the size of $\tilde{\alpha}$ is strictly greater than the one of $\tilde{\beta}$ or if both sizes of $\tilde{\alpha}$ and $\tilde{\beta}$ are equal and the composition $\underline{\alpha}$ is smaller than $\underline{\beta}$, using the reverse lexicographic order over compositions.*

Proposition 7.2.4. *The order defined over strict monotonic trees is a total order.*

The result is direct since all possible cases according to the trees α and β for comparing them are explored.

Theorem 7.2.5. *The function* UnrankTree *is an unranking algorithm and calling it with the parameters n and a uniformly-sampled integer s in $\{0, \ldots, g_n - 1\}$ gives as output a uniform strict monotonic Schröder tree of size n .*

The correctness of the algorithm follows directly from the total order over the trees and Equation (7.1).

Theorem 7.2.6. *Once the pre-computations have been done, the function* UnrankTree *needs on average $\Theta(n)$ arithmetic operations on big numbers to construct a tree of size n.*

Proof. Let us assume that all binomial coefficients $(C_{n,\ell})_{0 \leq \ell \leq n}$ have been memorised and prove that, with this information stored, the complexity in terms of arithmetic operations is of order $\Theta(n)$. Note that if we only memorise the factorial numbers $(i!)_{0\leq i\leq n-1}$ the complexity is at most three times the complexity obtained when memorising the binomial coefficient and thus still of order $\Theta(n)$.

For all $n \geq 1$, we denote by a_n the number of arithmetic operations on big numbers that come from the loop in line 7 and the calls in lines 11 and 12, when building all trees of size n (i.e. we sum the number needed for each $r \in \{0,\ldots,C_{n,\ell}-1\}$). The exact value of arithmetic operations is $a_n + O(ng_n)$, because at each recursive call there is at most a constant number of operations that are not counted in a_n. We first analyse a_n: we have

$$a_n = \sum_{\ell=1}^{n-1} \binom{n-1}{\ell} \big((\min(\ell, n-1-\ell) - 1 + 2\ell)\, g_{n-\ell} + a_{n-\ell}\big).$$

In fact, for the terms with index ℓ, we are interested in the trees α of size n such that their corresponding tree $\tilde{\alpha}$ is of size $n-\ell$. Thus such trees α are counted by $\binom{n-1}{\ell} g_{n-\ell}$. And for each of them the factor $\min(\ell, n-1-\ell) - 1$ is the the number of operations needed for the unranking of the composition (we use the symmetry in the binomial coefficients), the factor 2ℓ is the number of multiplication and subtractions in the loop in line 7. Furthermore we have $a_1 = 0$. By taking an upper bound for the min function, we get that if $\bar{a}_1 = 0$, and, for all $n \geq 2$,

$$\bar{a}_n = \sum_{\ell=1}^{n-1} \binom{n-1}{\ell} \left(3\ell\, g_{n-\ell} + \bar{a}_{n-\ell}\right),$$

then $a_n \leq \bar{a}_n$ for all $n \geq 1$. Using similar calculations as in the proof of Proposition 4.3.3, we obtain an equation satisfied by the Borel transform of the series associated to $(\bar{a}_n)$:

$$2\left(\mathcal{B}\bar{A}(z)\right)' = \mathrm{e}^z \left(\mathcal{B}\bar{A}(z)\right)' + 3z\mathrm{e}^z \ \left(\mathcal{B}G(z)\right)'.$$

We thus deduce $\bar{a}_n \sim 3ng_n$, which concludes the proof. □

Theorem 7.2.7. *Once the pre-computations have been done, the function* UnrankTree *needs on average* (n^2) *arithmetic operations to construct a tree of size* n.

Proof. The proof of this comes when we see that the number of arithmetic operations needed for the unranking of the composition is n. When we were counting the arithmetic operations on big numbers we were not counting any operation when entering the *if* condition. But now, this counts for one operation. The number of calls to UnrankComposition(n, l, r) is bounded by n and for each call we do exactly one operation.

For all $n \geq 1$, we denote a_n the number of arithmetic operations that come from the loop in line 7 and the calls in lines 11 and 12, when building all trees of size n (i.e. we sum the number needed for each $r \in \{0,\ldots,C_{n,\ell}-1\}$). The exact value of arithmetic operations is $a_n + O(ng_n)$, because at each recursive call there is at most a constant number of operations that are not counted in a_n. Let $\bar{a}_1 = 0$, and, for all $n \geq 2$,

$$\bar{a}_n = \sum_{\ell=1}^{n-1} \binom{n-1}{\ell} \left(n\, g_{n-\ell} + \bar{a}_{n-\ell}\right),$$

It is immediate that for all $n \geq 1, \bar{a}_n \geq a_n$. We also get as in the previous

$$\bar{a}_n \underset{n\to\infty}{\sim} \alpha\, n^2\, g_n,$$

since the **EGF** of $\bar{A}(z)$ is

$$2\left(\mathcal{B}\bar{A}(z)\right)' = \mathrm{e}^z\, \left(\mathcal{B}\bar{A}(z)\right)' + z(z\mathrm{e}^z\, \left(\mathcal{B}G(z)\right)')'.$$

Which concludes the proof. □

Let us give some final remarks for this algorithm. In order to obtain a better time complexity for the implementation, we must handle an array of pointers to the leaves of the tree under construction so that the tree traversal is efficient. At each step ℓ, a leaf stored in the array is replaced by $n - \ell + 1$ leaves that must be stored in the array. An efficient way consists in reusing the cell from the replaced leaf, and the to append all other leaves at the end on the array. Thus, the most efficient traversal $\mathcal{T}$ of the leaves consists to the left right traversal of the array. But obviously this is not really a natural traversal for the tree. Thus in practice we use this efficient traversal $\mathcal{T}$.

Some simulations on several parameters are presented in Figure 7.3. The lines between the points represent the theoretical expected values that we have shown in Section 4.3.

In Figure 7.1, the figures show that the average number of arithmetic operations on big number is $\Theta(n)$ and they suggest that the average total number of arithmetic operations is $\Theta(n^2)$. We have shown

7.2.3 Strict monotonic general Schröder

In this section we exhibit an efficient way for the uniform sampling of the tree model using the evolution process.

Once again when r grows, the sequence $(f_{n-r})_r$ decreases extremely fast. Thus for the uniform random sampling, it will appear more efficient to read Equation (4.27) in the following way:

$$\begin{aligned} f_n = &\binom{n-1}{1}\, 2^{n-2}\, f_{n-1} + \sum_{i=1}^{2} \binom{n-2}{i}\, 2^{n-2-i}\, f_{n-2} \\ &+ \sum_{i=1}^{3} \binom{n-3}{i}\, 2^{n-3-i}\, \binom{2}{i-1}\, f_{n-3} + \cdots + f_1. \end{aligned} \tag{7.3}$$

Using the latter decomposition the algorithm can now be described as Algorithm 5. Let us define first the algorithm for unranking a binomial that we give in Algorithm 4. The idea is to that if we want to choose k elements from n disposable ones. The algorithm gives a rank to each possible set of elements and then we can return the element the r-th element from the $\binom{n}{k}$ possible ones.

Algorithm 5 is very similar to the one corresponding to strict monotonic trees. In fact this new one is a bit more involved than the previous one because of the recurrence formula for enumerating the tree. However both algorithm cores are very close. First the `While`

Algorithm 4 Reverse Lexicographic binomial Unranking

```
function UnrankBinomial($n, \ell, r$)
    $C$ :=UnrankComposition($n+1, \ell+1, r$)
    $r := [C[0]]$
    $cumul := C[0]$
    for $i$ from 1 to $length(C) - 2$ do
        $cumul := cumul + C[i]$
        $r.append(cumul)$
    return $r$
```

Algorithm 5 Strict Monotonic General Tree Unranking

```
 1: function UnrankTree($n, s$)
 2:     if $n = 1$ then
 3:         return the tree reduced to a single leaf
 4:     $\ell := 1$
 5:     $r := s$
 6:     $i := 1$
 7:     while $r >= 0$ do
 8:         $t := \binom{n-\ell}{i} \, 2^{n-\ell-i} \, \binom{\ell-1}{i-1}$
 9:         $r := r - t \cdot f_{n-\ell}$
10:         $i := i + 1$
11:         if $i > min(\ell, n - \ell)$ then
12:             $i := 1$
13:             $\ell := \ell + 1$
14:     if $i > 1$ then
15:         $i := i - 1$
16:     else
17:         $\ell := \ell - 1$
18:         $i := min(\ell, n - \ell)$
19:     $r := r + t \cdot f_{n-\ell}$
20:     $T$ :=UnrankTree($n - \ell, r \mod f_{n-\ell}$)
21:     $r := r // f_{n-\ell}$
22:     $B$ :=UnrankBinomial($n - \ell, i, r \mod \binom{n-\ell}{i}$)
23:     $r := r // \binom{n-\ell}{i}$
24:     $F := r // \binom{\ell-1}{i-1}$
25:     $C$ :=UnrankComposition($\ell, i, r \mod \binom{\ell-1}{i-1}$)
26:     Substitute in $T$, using traversal $\mathcal{T}$, the leaves selected with $B$ with internal nodes
27:     and new leaves according to $C$ and the other leaves are changed or not based on $F$
28:     return the tree $T$
```

The sequences $(f_\ell)_{\ell \leq n}$ and $(\ell!)_{\ell \in \{1,\ldots,n\}}$ have been pre-computed and stored.

loop allows to determine the values for ℓ, i and r. Then the recursive call is done using the adequate rank $r \mod f_{n-\ell}$. The last lines of the algorithm (for 21 to 27) are necessary to modify the tree T of size $n-\ell$ that has just been built. In line 22 we determine which leaves T will be substituted by internal nodes (of arity at most 2) with new leaves. It is based on the unranking of combinations that is very close to the unranking of compositions. Then for the other leaves that are either kept as they are of replaced by unary internal nodes attached to a leaf we use the integer F seen as a $n-\ell-i$-bit integer: if the bit $\#s$ is 0 then the corresponding leaf is kept, and if it is 1 then the leaf is substituted. And finally the composition unranking allows to determine how many leaves are attached to the nodes selected with B.

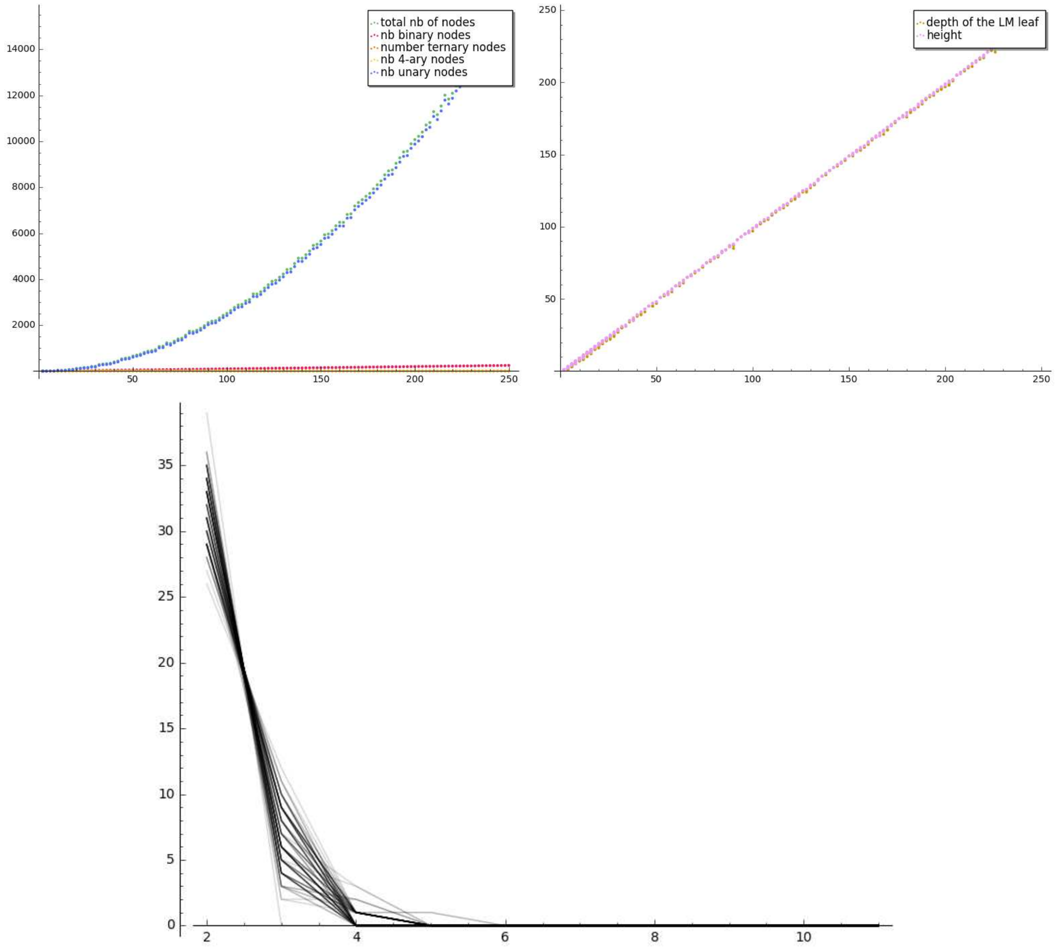


Figure 7.4: Simulation of different parameters on *general strict monotonic Schröder trees*. (left) The number of d-ary nodes and the number of iteration steps. (right) The height and the depth of the (LM) leftmost leaf. The sizes of the sampled trees range from 1 to 250. (down) Root degree, the lines gets darker as n moves from 2 to 100, with 40 trees sampled for each size.

Theorem 7.2.8. *The function* UnrankTree *is an unranking algorithm and calling it with the parameters n and a uniformly-sampled integer s in $\{0, \dots, f_n - 1\}$ gives as output a uniform strict monotonic general tree of size n .*

The correctness of the algorithm follows directly from the total order over the trees deduced from the decomposition Equation (7.3).

Theorem 7.2.9. *Once the pre-computations have been done, the function* UnrankTree *needs in average $\Theta(n)$ arithmetic operations on big numbers to construct a tree of size n.*

The proof for this theorem is analogous to the one for Theorem 7.2.6 corresponding to the complexity of the tree builder for strict monotonic Schröder trees.

We have depicted some simulations of some parameters in Figure 7.4. These simulations agree with the theoretical results found in Section 4.4. In fact, we unary nodes are allowed many of them appear at each iteration step. Since they do not add the final size of the tree their number can grow quadratically. This also affects the height that becomes linear rather than logarithmic as we saw with the two other models.

7.3 General Model uniform random generation

In order to have exact size uniform random generation, we need to work with the general recurrence Equation (5.2) of the *evolution process* Definition 5.2.1.

We start this section by recalling Definition 5.4.1 seen in Chapter 5:

Definition 7.3.1. *We denote by $A_{n,r,k,\phi}$ the set of ordered multisets with elements in $\mathbb{N}$ such that for each ordered multiset $a = [a_1, \dots, a_l] \in A_{n,k,r,\phi}$:*

- *We have that $a_1 + \cdots + a_l = k$.*
- *The elements $a_1, \dots, a_l$ are ordered decreasingly.*
- *$|a| \in r$ and $|a| \leq (n-k)$.*
- *$\forall i, 1 \leq i \leq l, [z^{a_i+1}]\phi(z) > 0$.*

where $|a|$ represents the size of the list a.

The elements of the set $A_{n,k,r,\phi}$ are used in the main recurrence of the *evolution process*. It means that for unranking method to work and even for enumeration purposes we need to be able to generate these elements. We recall the main recurrence of the process Equation (5.2) seen in Section 5.4:

$$B_n = \begin{cases} 0 & \text{if } n < m \\ 1 & \text{if } n = m \\ \sum\limits_{k=1}^{n-1} \left(\sum\limits_{a \in A_{n,k,r,\phi}} \frac{(n-k)!}{(n-k-|a|)!} \prod\limits_{i=0,u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!} \right) B_{n-k} & \text{if } n > m \end{cases} \tag{7.4}$$

As we will see it is possible to generate a super set of $A_{n,k,r,\phi}$, namely $A_{n,k,\mathbb{N}^*,\phi}$. We just take the set of allowed repetitions to be all possible repetitions. Therefore it is immediate that

$$A_{n,k,r,\phi} \subset A_{n,k,\mathbb{N}^*,\phi}.$$

It is then possible to filter the elements $A_{n,k,\mathbb{N}^*,\phi}$ to produce exactly the desired set $A_{n,k,r,\phi}$. The set $A_{n,k,\mathbb{N}^*,\phi}$ is related to restricted *integer partitions* through a bijection with an intermediate object. It will be easier to add an element for the proofs of the bijections to add an element a_0 in the elements of $A_{n,k,\mathbb{N}^*,\phi}$.

For simplicity we will denote and $A_{n,k,\phi}$ for $A_{n,k,\mathbb{N}^*,\phi}$. In fact:

$$A_{n,k,\phi} = \bigcup_{i=1}^{\infty} A_{n,k,i,\phi}$$

A related set of objects $\mathbf{A}_{n,k,r,\phi}$ that we present hereafter will be used in order to generate the set $A_{n,k,r,\phi}$. This set $\mathbf{A}_{n,k,r,\phi}$ is closely related to *integer partitions* for which plenty of algorithms exist to generate them.

Definition 7.3.2. *We denote by $\boldsymbol{A}_{n,r,k,\phi}$ the set of ordered multisets with elements in $\mathbb{N}^*$ such that for each ordered multiset $a = [a_1, \dots, a_l] \in \boldsymbol{A}_{n,k,r,\phi}$:*

- $a_0 = n - k$.
- *We have that* $a_1 + \cdots + a_l = k$.
- *The elements* $a_1, \dots, a_l$ *are ordered decreasingly.*
- $|a| \in r$ *and* $|a| \leq (n - k)$.
- $\forall i, 1 \leq i \leq l, [z^{a_i+1}]\phi(z) > 0$.

where $|a|$ represents the size of the list a. we denote by $\boldsymbol{A}_{n,k,\phi}$

$$\boldsymbol{A}_{n,\phi} = \bigcup_{i=1}^{n-1} \boldsymbol{A}_{n,i,\phi}$$

The same definitions can be done with $A_{n,\phi}$ and $A_{n,k,\phi}$.

The main difference between $\mathbf{A}_{n,r,k,\phi}$ and $A_{n,r,k,\phi}$, is the fact that the former does not contain zeros. The former incorporate also an additional element a_0, which is useful for the proofs of the bijection.

We remind that the zeros in $A_{n,r,k,\phi}$ account for building varieties of trees where we have unary nodes. In fact $\mathbf{A}_{n,r,k,\phi}$ in essence in the set of trees where unary nodes are not allowed. In the end of the generation process, once we can generate $\mathbf{A}_{n,r,k,\phi}$, it will be simple to modify it and obtain $A_{n,r,k,\phi}$. We take each element of $\mathbf{A}_{n,r,k,\phi}$, and some zeros to it and check if the new element belongs to $A_{n,r,k,\phi}$.

7.3.1 Generating the elements of $\mathbf{A}_{n,\phi}$ from Integer partitions

We will need the same operations that have been defined on $A_{n,k,r,\phi}$. Therefore we use a slightly modified version of Definition 5.4.2:

Definition 7.3.3. *Let a be an ordered list of integers. We define the* maximum function *max to be the function that maps $a\backslash\{a_0\}$ to its greatest element. We define the* occurrences function *to be the one such that $occ(a) = [u_0, \dots, u_{max(a)}]$ where u_i is the number of elements in $a\backslash\{a_0\}$ equal to i. For example, when $a = [4, 3, 1, 1]$, $occ(a) = [0, 2, 0, 1]$.*

From this definition we can immediately write a pseudo-code for the *occ* function, see Algorithm 6.

The function *occ* serves for computing the product of inside the recurrence. We define now the restricted class of *integer partitions* that we will use:

Definition 7.3.4. *Let $P_{n,k,\phi}$ be the set of integer partitions of n: such that for each $p = [p_0, \dots, p_m] \in P_{n,k,\phi}$:*

Algorithm 6 The *occ* function

```
function $occ(a$ : an (ordered list))
    $l := copy(a)$
    $l.pop(0)$
    $res := []$
    for $i$ from 0 to $length(l)$ do
        $res[l[i]] := res[l[i]] + 1$
    return $res$
```

$copy()$ copies the element of a list to make complete independent copy of the list.

- *We take the decreasing ordering as a canonical order.*
- $p_0 = k$.
- *for all* $1 \leq i \leq m$, *if* $p_i - p_{i+1} > 0 \implies [z^{i+1}]\phi(z) > 0$.

We define

$$P_{n,\phi} = \bigcup_{k=1}^{n-1} P_{n,k,\phi}.$$

Our main objective is to show an equivalence (a bijection) between $\mathbf{A}_{n,\phi}$ and $P_{n,\phi}$. Since integer partitions are a well known and well studied class of objects. We can use efficient generation algorithms and adapt them for our problem.

We define the following class of object that will be the intermediate class between $\mathbf{A}_{n,\phi}$ and $P_{n,\phi}$. This intermediate class will help proving the bijection.

Definition 7.3.5. $V_{n,k,\phi}$ *the set of ordered multisets with elements in* $\mathbb{N}$ *such that for each ordered multiset* $v = [v_0, v_1, \ldots, v_l] \in V_{n,k,\phi}$*:*

- $v_0 = n, v_1 = k, v_l = 0$.
- *The sequence is strictly decreasing that is* $\forall, 0 \leq i \leq l-1, v_i < v_{i-1}$.
- *The spacings (difference between two consecutive elements) is always decreasing, that is* $\forall i, 0 \leq i \leq l-2, v_i - v_{i+1} \geq v_{i+1} - v_{i+2}$.
- $\forall i, 1 \leq i \leq l-2$, *if* $v_i - 2v_{i+1} + v_{i+2} > 0 \implies [z^{i+1}]\phi(z) > 0$.

For instance, for $\phi(z) = \frac{z}{1-z}$, $[6,4,3,0] \notin V_{6,4,\phi}$ *while* $[6,4,2,0] \in V_{6,4,\phi}$.

We define

$$V_{n,\phi} = \bigcup_{k=1}^{n-1} V_{n,k,\phi}.$$

Finally, we define $V_{n,k,i,\phi} = \{v = [v_0, v_1, v_2, \ldots, v_l] \mid v \in V_{n,k,\phi} \wedge v_2 = i\}$.

In Table 7.1, we depict an example of the bijection to be proven between these three classes of objects. We see on that example that the differences between the elements of $V_{n,\phi}$ are the elements of the *integer partition*. Now we will show the correspondence between these classes. The diagram in Figure 7.5 sums up the results.

$$\begin{tikzcd}\mathbf{A}_{n,\phi} \arrow[r, shift left, "\mathcal{M}"] & V_{n,\phi} \arrow[l, shift left, "\mathcal{G}"] \arrow[r, shift left, "\mathcal{N}"] & P_{n,\phi} \arrow[l, shift left, "\mathcal{F}"]\end{tikzcd}$$

Figure 7.5: The bijections between $\mathbf{A}_{n,\phi}, V_{n,\phi}, P_{n,\phi}$.

Definition 7.3.6. *The function $\mathcal{M}$ is a mapping from $\boldsymbol{A}_{n,\phi}$ to $V_{n,\phi}$ such that $\forall a \in \boldsymbol{A}_{n,\phi}$, if $a = [a_1, \dots, a_l]$, $occ(a) = [u_0, \dots, u_{max(a)}]$, and $m = max(a)$ we have:*

- $v_0 = sum(a)$.
- $v_m = u_m, v_{m+1} = 0$
- $\forall i, 1 \leq i \leq m-1, v_{m-i} = u_{m-i} + 2v_{m-i+1} - v_{m-i+2}$.

$V_{n,\phi}$	$\mathbf{A}_{n,\phi}$	$P_{n,\phi}$
[7, 1, 0]	[6, 1]	[6, 1]
[7, 2, 0]	[5, 1, 1]	[5, 2]
[7, 2, 1, 0]	[5, 2]	[5, 1, 1]
[7, 3, 0]	[4, 1, 1, 1]	[4, 3]
[7, 3, 1, 0]	[4, 2, 1]	[4, 2, 1]
[7, 3, 2, 1, 0]	[4, 3]	[4, 1, 1, 1]
[7, 4, 1, 0]	[3, 2, 1, 1]	[3, 3, 1]
[7, 4, 2, 0]	[3, 2, 2]	[3, 2, 2]
[7, 4, 2, 1, 0]	[3, 3, 1]	[3, 2, 1, 1]
[7, 4, 3, 2, 1, 0]	[3, 4]	[3, 1, 1, 1, 1]
[7, 5, 3, 1, 0]	[2, 3, 2]	[2, 2, 2, 1]
[7, 5, 3, 2, 1, 0]	[2, 4, 1]	[2, 2, 1, 1, 1]
[7, 5, 4, 3, 2, 1, 0]	[2, 5]	[2, 1, 1, 1, 1, 1]
[7, 6, 5, 4, 3, 2, 1, 0]	[1, 6]	[1, 1, 1, 1, 1, 1, 1]

Table 7.1: Table of values for $V_{7,\phi}, \mathbf{A}_{7,\phi}$ and $P_{7,\phi}$ with $\phi(z) = \frac{z}{1-z}$ and $r = \mathbb{N}^*$

Proposition 7.3.7. *The mapping $\mathcal{M}$ is injective.*

Proof. The proof is postponed to Section 7.5.1. □

We define now what we call the num function. This function will be used in the mapping $\mathcal{G}$. In essence num counts the number

Definition 7.3.8. *We define $num(v, n)$ by the following. Let $m = last(v)$, where $last(v)$ is the last index i such that $v_i > 0$. Then $num(v, n)$ is defined as the ordered list a such that*

- $a_0 = n - (\sum\limits_{i=0}^{|v|-1} iv_i)$.
- *For i going from 0 to $m-1$, append v_i occurrences of i to the list a.*

For example, $d = [0,2,0,1], n = 9$, $num(d,9) = [4,3,1,1]$.

Remark 7.3.9. *We have that $\forall a \in \boldsymbol{A}_{n,\phi}, num(occ(a),n) = a$.*

From Figure 7.5, we see that for our application. Since we will generate $\mathbf{A}_{n,\phi}$ from $\mathcal{P}_{n,\phi}$, we will give the pseudo-codes of the mappings $\mathcal{G}$ and $\mathcal{F}$ because they go to the right direction.

Definition 7.3.10. *The function $\mathcal{G}$ is a mapping from $V_{n,\phi}$ to $\boldsymbol{A}_{n,\phi}$ such that $\forall v \in V_{n,\phi}$, if $v = [v_0,,\ldots,v_l]$. We construct an new ordered list b having:*

- $b_{l-2} = v_{l-1}$.
- $\forall i, 1 \leq i \leq l-3, b_i = v_i - 2v_{i+1} + v_{i+2}$.

Then the result of the mapping is a new list $a = [v_0 - v_1] + num(b,n)$. Where "+" is the operation of concatenation of two ordered lists.

The pseudo-code of $\mathcal{G}$ is presented in Algorithm 7. In the following Propositions we will prove that $\mathcal{N}$ and $\mathcal{F}$ form a bijection.

Algorithm 7 The mapping $\mathcal{G}$

```
function $v\_to\_arities(v$ : (ordered list))
    $res := [v[0] - v[1]]$
    for $i$ from 1 to $length(v)$ do
        if $length(v) - 1 - i \geq 2$ then
            $k := v[i] - 2v[i+1] + v[i+2]$
            for $j$ from 0 to $k-1$ do
                $res.insert(i)$
        else if $length(v) - 1 - i = 1$ then
            $k := v[i] - 2v[i+1]$
            for $j$ from 0 to $k-1$ do
                $res.append(i)$
        else
            $k := v[i]$
            for $j$ from 0 to $k-1$ do
                $res.append(i)$
    $x := res.pop(0)$
    $res.sort\_reverse\_order()$
    $res.insert(0,x)$
    return $res$
```

$pop(i)$ function, removes i-th element from a list and returns it.

$insert(i,e)$ function, insert the element e at the i-th position of the list.

$sort\_reverseorder()$ function, sorts a list of element in a decreasing order.

Proposition 7.3.11. *The mapping $\mathcal{G}$ is injective.*

The proof is postponed to Section 7.5.1.

Proposition 7.3.12. $\forall a \in \boldsymbol{A}_{n,\phi}, \mathcal{G}(\mathcal{M}(a)) = a.$

Proof. The proof is postponed to Section 7.5.1. □

Remark 7.3.13. *The mapping $\mathcal{G}$ is the inverse mapping of $\mathcal{M}$*

From these previous Propositions the bijection between $\mathbf{A}_{n,\phi}$ and $V_{n,\phi}$ follows immediately.

Theorem 7.3.14. *The mapping $\mathcal{M}$ and its inverse $\mathcal{G}$ is a bijection between $\boldsymbol{A}_{n,\phi}$ and $V_{n,\phi}$.*

Proof. The result follows directly from Proposition 7.3.7, Proposition 7.3.11 and Proposition 7.3.12. □

Now we show the second bijection between $\mathcal{P}_{n,\phi}$ and $\mathcal{V}_{n,\phi}$. We will follow the same steps as we did for the first bijection. We give the definitions of the mappings $\mathcal{N}$ and $\mathcal{F}$. We give the pseudo-code of $\mathcal{F}$ since it is the direction of interest to us for the generation. Finally, we prove the bijection.

Definition 7.3.15. *The function $\mathcal{N}$ is a mapping from $V_{n,\phi}$ to $P_{n,\phi}$. If $v = [v_0, \dots, v_m] \in V_{n,\phi}$, and $p = [p_0, \dots, p_{m-1}]$. We define $\forall i, 0 \leq i \leq m-1, p_i = v_i - v_{i+1}$. Then $\mathcal{N}(v) = p$.*

Definition 7.3.16. *The function $\mathcal{F}$ is a mapping from $P_{n,\phi}$ to $N_{n,\phi}$. If $p = [p_0, \dots, p_m] \in p_{n,\phi}$, and $v = [v_0, \dots, v_{m+1}]$. We define $v_0 = sum(p)$ $\forall i, 0 \leq i \leq m, v_{i+1} = v_i - p_i$. Then $\mathcal{F}(p) = v$.*

From the definition of $\mathcal{F}$ it is possible to directly write a pseudo-code for it that we present in Algorithm 8. Both definitions are simple. In fact, the elements of $\mathcal{P}_{n,\phi}$ are just the differences between the elements of $V_{n,\phi}$.

Algorithm 8 The mapping $\mathcal{F}$

```
function $part\_to\_v$($p$ : an integer partition (ordered list))
    $n := sum(p)$
    $res := [n]$
    for $i$ from 0 to $length(p) - 1$ do
        $res.append(res[i] - p[i])$
    return $res$
```

The $append(e)$ function appends the element e to the end of an existing list.

Proposition 7.3.17. *The mapping $\mathcal{N}$ is injective.*

The proof is in Section 7.5.1.

Proposition 7.3.18. $\forall v = [v_0, \dots, v_m] \in V_{n,\phi}$, $\mathcal{F}(\mathcal{N}(v)) = v.$

The proof is in Section 7.5.1.

Remark 7.3.19. *$\mathcal{F}$ is the inverse function of $\mathcal{V}$*

The bijection follows directly from the previous Propositions.

Theorem 7.3.20. *The mapping $\mathcal{V}$ and its inverse $\mathcal{F}$ is a bijection between $V_{n,\phi}$ and $P_{n,\phi}$*

Proof. The result follows from Proposition 7.3.17 and Proposition 7.3.18. □

7.3.2 Sampling algorithm

We can now find a new recurrence to our general problem by using the bijections. We first need to show a small result. In order to make our random generation Algorithm we needed to generate the set $\mathbf{A}_{n,k,r,\phi}$. The problem has now been simplified to the one of generating *integer partitions* and then use the mappings $\mathcal{F}$ and $\mathcal{G}$ defined in Definition 7.3.10 and Definition 7.3.16 to get the set $\mathbf{A}_{n,k,\phi}$ and filter it to get $\mathbf{A}_{n,k,r,\phi}$.

Algorithm 9 The inner product of the recurrence : $\frac{(n-k)!}{(n-k-|a|)!}\prod_{i=0,u_i\neq 0}^{|occ(a)|}\frac{\phi_{i+1}^{u_i}}{u_i!}$

```
function $prod(a$ : an (ordered list), $n$ :integer, $k$ :integer, $\phi$ : function $\mathbb{N}^* \to \mathbb{N}^*)$
    $l := occ(a)$
    $p := 1$
    for $i$ from 0 to $length(l) - 1$ do
        if $l[i] \neq 0$ then
            $p := p \times \frac{\phi_{i+1}^{l[i]}}{(l[i])!}$
    $p := p \times \frac{(n-k)!}{(n-k-length(a)+1)!}$
    return $p$
```

We need now to make the algorithm that generates all partitions of size n and filter them according to $\phi(z)$ and the set r of allowed repetitions to generate $\mathbf{A}_{n,r,\phi}$. Finally, in order to generate $A_{n,r,\phi}$, we need to see where it is possible to add 0 elements since the set $\mathbf{A}_{n,r,\phi}$ has no zeros. For example we see that in Table 7.1, unary nodes are allowed since $\phi_1 = 1$. But no element in $\mathbf{A}_{7,z/(1-z)}$ contains zeros.

Therefore, in order to generate $\mathcal{A}_{7,z/(1-z)}$ from $\mathbf{A}_{7,z/(1-z)}$, we take each element in $\mathbf{A}_{7,z/(1-z)}$, and make new sets with zeros as long as the condition $|a| \leq (n-k)$ is met.

Algorithm 10 Check if the node arities are accepted (i.e all arities belong to $\phi(z)$).

```
function $check\_node\_arities(a$ : an (ordered list) , $\phi$ : function $\mathbb{N}^* \to \mathbb{N}^*)$
    for $i$ from 1 to $length(a) - 1$ do
        if $\phi(a[i] + 1) = 0$ then
            return 0
    return 1
```

Algorithm 12, generates exactly the set $A_{n,r,\phi}$ of elements required for the recurrence Equation (7.4). Since it generates $\mathbf{A}_{n,r,\phi}$ through the mappings $\mathcal{F}$ and $\mathcal{G}$ defined by the functions $part\_to\_v$ and $v\_to\_arities$. Once this is done, for each element of the resulting set we try to add zeros (i.e unary nodes are allowed) to see if they are allowed in terms of arities through ϕ and if they are allowed in terms of the number of nodes that can grow in one iteration step through the set r.

Algorithm 11 Check if the number of repetitions is accepted (i.e the number of node added at this step is in r).

```
function $check\_nb\_repetitions(a$ : an (ordered list), $r$ : function $\mathbb{N}^* \to \{0,1\})$
    if $r(length(a)-1) = 1$ then
        return $1$
    else
        return $0$
```

Algorithm 12 Generate $A_{n,r,\phi}$

```
function $generate\_A(n$ : integer, , $\phi$ : function $\mathbb{N}^* \to \mathbb{N}^*$, $r$ : function $\mathbb{N}^* \to \{0,1\})$
    $res := []$
    $l := generate\_partitions(n)$
    for $e \in l$ do
        $e.sort\_reverse\_order()$
        $v := part\_to\_v(e)$
        $a := v\_to\_arities(v)$
        if $check\_node\_arities(a, \phi) = 1$ then
            if $check\_nb\_repetitions(a, r) = 1$ then
                $res.append(a)$
            if $\phi(1) > 0$ then
                for $i$ from $length(a) - 1$ to $a[0] - 1$ do
                    $tmp := a.copy()$
                    $tmp.append(0)$
                    if $check\_nb\_repetitions(tmp) = 1$ then
                        $res.append(tmp)$
    return $res$
```

$generate\_partitions(n)$ function, returns a list of all integer partitions of n. Each element is an ordered list of integers

If we let:

$$t_{n,k} = \sum_{a \in A_{n,k,r,\phi}} \frac{(n-k)!}{(n-k-|a|)!} \prod_{i=0,u_i \neq 0}^{|occ(a)|} \frac{\phi_{i+1}^{u_i}}{u_i!},$$

It is possible to compute $t_{n,k}$ from $\mathbf{A}_{n,r,\phi}$, we only need to go through $\mathbf{A}_{n,r,\phi}$ and keep the elements that starts with $n-k$, then sum the inner products using Algorithm 9.

Theorem 7.3.21. *The pre-computations can be done in* $O\left(\mathrm{e}^{\sqrt{n}}\right)$ *arithmetic operations.*

Proof. The pre-computations involve computing the sequence B and the set $A_{n,r,\phi}$. As we saw the set $A_{n,r,\phi}$ is built from *integer partitions* of n which have complexity:

$$O\left(\frac{\mathrm{e}^{\sqrt{n}}}{n}\right),$$

in term of the number of arithmetic operations, see [**YKKN07**] for more details. Each integer partition is then analysed and we try to add zeros to it which necessitates at worst case n operations. After storing the values of the set, the sequence B can be computed in $O\left(n^2\right)$. □

Algorithm 13 Sampling

```
function SampleTree($n : integer, \phi$ : function $\mathbb{N}^* \to \mathbb{N}^*$, $r$ : function $\mathbb{N}^* \to \{0,1\}$)
    if $n = 1$ then
        return the tree reduced to a single leaf
    $s := randint(0, B[n])$
    $k := 1$
    while $s >= 0$ do
        $s := s - t_{n,k} \times B[n-k]$
        $k := k + 1$
    $k := k - 1$
    $T$ :=SampleTree($n - k, \phi, r$)
    $weights := []$
    for $e \in A_{n,k,r,\phi}$ do
        $weights.append(prod(e, n, k, \phi)/t_{n,k}$
    $ar$ :=choice($A_{n,k,r,\phi}$,$weights$)
    $ar.pop(0)$
    $col := []$
    for $i$ from 0 to $length(ar)$ do
        $col.append(randint(1, \phi(i+1)))$
    $ls := UnrankBinomial(n - k, len(ar))$
    Substitute in $T$, using traversal $\mathcal{T}$, the leaves selected with in $ls$ with internal nodes of the degrees of the list $ar$ with colours selected in $col$
    return the tree $T$
```

The sequences $(B_\ell)_{\ell \leq n}$ and $A_{\ell,k,r,\phi}$ for $\ell < n, 1 \leq k \leq n-1$ have been pre-computed and stored.
$choice(list, weights)$ chooses an element from a list according to the weights of the elements.

Algorithm 13 is based on the recursive generation method. However, we are faced with a problem in the choice function. Each element of the set $A_{n,r,\phi}$ has a weight, because the same configuration of new nodes at an iteration step can be put on different leaves. Either, we can store the weight of an element as a repetition of this element same element and make a new set and finally sample uniformly from it. If we store exactly $A_{n,r,\phi}$. Then there is a need to sample from this set according the weights of each element.

The variable ar, contains the new nodes that will be added to the tree at this ietration step. The the variable col, contains the color of each new node in ar. Finally, the variable ls, serves to choose wich leaves will get expanded at the current iteration step.

Theorem 7.3.22. *The function* SampleTree *operates in* $O\left(\mathrm{e}^{\sqrt{n}}\, n^2\right)$ *arithmetic operations.*

Proof. The number of calls of SampleTree is bounded by $n-1$. The UnrankBinomial and UnrankComposition functions need $O(n)$ arithmetic operations as it has been seen in Section 7.2.2. Then the choice function is $O\left(\mathrm{e}^{\sqrt{n}}\right)$, since it needs to go through all elements of $A_{n,k,r,\phi}$. □

7.4 Conclusion

For the three classes of *increasing Schröder trees*, we saw that some simplifications occur in the main recurrence, and thus we can take advantage of these situations to design fast uniform samplers. For *increasing Schröder trees*, the uniform sampler it very fast, linear in time, so we can go to big sizes of order 20000 in some seconds. For *strict monotonic Schröder trees*, we can go to sizes of order 1000 in few seconds also after having done pre-computations. For the third model *general monotonic Schröder trees*, the size of the numbers in the main recurrence are of order n^2. Even if, the average number of arithmetic operations on big numbers is $\Theta(n)$, the intrinsic complexity of the model is higher. For this model we can go to sizes up to 500 in few seconds.

Finally, the generam model relies on generating and storing *restricted integer partitions* which can not be done in polynomial time. For the general case we can go to sizes of order 50 in few seconds after the pre-computations have been made.

Bibliography

[Ald96] David John Aldous. Probability Distributions on Cladograms. In *Random Discrete Structures*, pages 1–18. Springer New York, 1996.

[ALSU06] Alfred Vaino Aho, Monica Sin-Ling Lam, Ravi Sethi, and Jeffrey David Ullman. *Compilers: Principles, Techniques, and Tools (2nd Edition)*. Addison-Wesley Longman Publishing Co., Inc., USA, 2006.

[AS95] Laurent Alonso and René Schott. *Random Generation of Trees*. Springer US, Boston, MA, 1995.

[BBD18] Maciej Bendkowski, Olivier Bodini, and Sergey Dovgal. Polynomial tuning of multiparametric combinatorial samplers. In *2018 Proceedings of the Fifteenth Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, pages 92–106. SIAM, 2018.

[BBHT17] Axel Bacher, Olivier Bodini, Hsien-Kuei Hwang, and Tsung-Hsi Tsai. Generating Random Permutations by Coin Tossing: Classical Algorithms, New Analysis, and Modern Implementation. *ACM Trans. Algorithms*, 13(2):24:1—-24:43, 2017.

[BDF$^+$16] Olivier Bodini, Matthieu Dien, Xavier Fontaine, Antoine Genitrini, and Hsien-Kuei Hwang. Increasing Diamonds. In *Theoretical Informatics - 12th Latin American Symposium (LATIN)*, pages 207–219, 2016.

[BDGP17] Olivier Bodini, Matthieu Dien, Ant Genitrini, and Frédéric Peschanki. The Ordered and Colored Products in Analytic Combinatorics: Application to the Quantitative Study of Synchronizations in Concurrent Processes. In *In Proceedings of the Meeting on Analytic Algorithmics and Combinatorics (ANALCO)*. SIAM, 2017.

[BDGV18] Olivier Bodini, Matthieu Dien, Antoine Genitrini, and Alfredo Viola. Beyond series-parallel concurrent systems: the case of arch processes. In *29th International Meeting on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms (AOFA)*, pages 14:1—-14:14, 2018.

[BDMdlS08] Nicolas Broutin, Luc Devroye, Erin McLeish, and Muriel de la Salle. The height of increasing trees. *Random Struct. Algorithms*, 32(4):494–518, 2008.

[Ben73] Edward A. Bender. Central and local limit theorems applied to asymptotic enumeration. *Journal of Combinatorial Theory, Series A*, 15(1):91–111, 1973.

[Ben74] Edward A. Bender. Asymptotic Methods in Enumeration. *SIAM Review*, 16(4):485–515, 1974.

[BFS92] François Bergeron, Philippe Flajolet, and Bruno Salvy. Varieties of increasing trees. In *CAAP*, pages 24–48, 1992.

[BGGW20] Olivier Bodini, Antoine Genitrini, Bernhard Gittenberger, and Stephan Wagner. On the number of increasing trees with label repetitions. *Discrete Mathematics*, 343(8):111722, 2020.

[BGMN20] Olivier Bodini, Antoine Genitrini, Cécile Mailler, and Mehdi Naima. Strict monotonic trees arising from evolutionary processes: combinatorial and probabilistic study. 2020.

[BGN19] Olivier Bodini, Antoine Genitrini, and Mehdi Naima. Ranked Schröder Trees. In *2019 Proceedings of the Sixteenth Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, pages 13–26. SIAM, 2019.

[BGNS20] Olivier Bodini, Antoine Genitrini, Mehdi Naima, and Alexandros Singh. Families of Monotonic Trees: Combinatorial Enumeration and Asymptotics. In *15th International Computer Science Symposium in Russia (CSR)*, pages 155–168, 2020.

[BGP16] Olivier Bodini, Antoine Genitrini, and Frédéric Peschanski. A Quantitative Study of Pure Parallel Processes. *Electronic Journal of Combinatorics*, 23(1):P1.11, 2016.

[Big79] Norman Linstead Biggs. The roots of combinatorics. *Historia Mathematica*, 6(2):109–136, 1979.

[BL77] Bill P Buckles and Matthew Lybanon. Algorithm 515: Generation of a Vector from the Lexicographical Index [G6]. *ACM Trans. Math. Softw.*, 3(2):180–182, 1977.

[Bli87] Johann Blieberger. Monotonically labelled Motzkin trees. *Discrete Applied Mathematics*, 18(1):9–24, 1987.

[BMLMN15] Mireille Bousquet-Mélou, Markus Lohrey, Sebastian Maneth, and Eric Noeth. XML Compression via Directed Acyclic Graphs. *Theory of Computing Systems*, 57(4):1322–1371, 2015.

[BO99] Carl M Bender and Steven A Orszag. *Advanced Mathematical Methods for Scientists and Engineers: Asymptotic methods and perturbation theory*. Springer, 1999.

[Bod10] Olivier Bodini. Autour de la génération aléatoire sous modèle de Boltzmann. In *Habilitation memoir*. Université Pierre et Marie Curie, 2010.

[Bol01] Béla Bollobás. *Random Graphs*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2 edition, 2001.

[Bón12] Miklós Bóna. *Combinatorics of permutations*. CRC Press, 2012.

[BP10] Olivier Bodini and Yann Ponty. Multi-dimensional Boltzmann sampling of languages. *Proceedings of the 21st International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms (AOFA)*, pages 49–64, 2010.

[Bro11] Lyle D Broemeling. An account of early statistical inference in Arab cryptology. *American Statistician*, 65(4):255–257, 2011.

[BRS12] Olivier Bodini, Olivier Roussel, and Michèle Soria. Boltzmann samplers for first-order differential specifications. *Discrete Applied Mathematics*, 160(18):2563–2572, 2012.

[Car59] Leonard Carlitz. Eulerian Numbers and Polynomials. *Mathematics Magazine*, 32(5):247, 1959.

[CCG18] Brigitte Chauvin, Julien Clément, and Danièle Gardy. *Arbres pour l'algorithmique*, volume 83. Springer, 2018.

[CCLR09] Stein Clifford, Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms, 3rd edition*. MIT press, 2009.

[Cha08] Charalambos A. Charalambides. Enumerative combinatorics. *ACM SIGACT News*, 2008.

[CHY00] Hua Huai Chern, Hsien-Kuei Hwang, and Yeong-Nan Yeh. Distribution of the number of consecutive records. *Random Structures & Algorithms*, 17(3-4):169–196, 2000.

[CMST17] Jacek Cichoń, Abram Magnery, Wojciech Szpankowskiz, and Krzysztof Turowskix. On symmetries of non-plane trees in a non-uniform model. *14th Workshop on Analytic Algorithmics and Combinatorics, (ANALCO)*, pages 156–163, 2017.

[Com12] Louis Comtet. *Advanced Combinatorics: The Art of Finite and Infinite Expansions*. Springer Science & Business Media, 2012.

[Con12] John Horton Conway. *The book of numbers*. Springer Science & Business Media, 2012.

[CPQ96] Douglas E. Critchlow, Dennis K. Pearl, and Chunlin Qian. The triples distance for rooted bifurcating phylogenetic trees. *Systematic Biology*, 45(3):323–334, 1996.

[CT05] Thomas M Cover and Joy A Thomas. *Elements of Information Theory*. John Wiley & Sons, 2005.

[Dev90] Luc Devroye. On the Height of Random m-ary Search Trees. *Random Struct. Algorithms*, 1(2):191–204, 1990.

[DFLS04] Philippe Duchon, Philippe Flajolet, Guy Louchard, and Gilles Schaeffer. Boltzmann samplers for the random generation of combinatorial structures. *Combinatorics Probability and Computing*, 13(4-5):577–625, 2004.

[DGH] Cyann Donnot, Antoine Genitrini, and Yassine Herida. Unranking Combinations Lexicographically: an efficient new strategy compared with others.

[Die17] Matthieu Dien. *Concurrent process and combinatorics of increasingly labeled structures : quantitative analysis and random generation algorithms. (Processus concurrents et combinatoire des structures croissantes : analyse quantitative et algorithmes de génération aléatoi*. PhD thesis, Pierre and Marie Curie University, Paris, France, 2017.

[DIMR09] Michael Drmota, Alex Iksanov, Martin Moehle, and Uwe Roesler. A limiting distribution for the number of cuts needed to isolate the root of a random recursive tree. *Random Struct. Algorithms*, 34(3):319–336, 2009.

[Drm03] Michael Drmota. An analytic approach to the height of binary search trees II. *ACM*, 50(3):333–374, 2003.

[Drm09] Michael Drmota. *Random trees: An interplay between combinatorics and probability*. Springer Science & Business Media, 2009.

[Dur06] Rick Durrett. *Random Graph Dynamics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2006.

[ER59] Paul Erdös and Alfréd Rényi. On random graphs. *Publicationes mathematicae*, 6(26):290–297, 1959.

[EY25] F. Y. E. and G. Udny Yule. A Mathematical Theory of Evolution Based on the Conclusions of Dr. J. C. Willis, F.R.S. *Journal of the Royal Statistical Society*, 88(3):433, 1925.

[Fel03] Joseph Felsenstein. *Inferring phylogenies*. Sinauer Associates, 2003.

[FFP07] Philippe Flajolet, Éric Fusy, and Carine Pivoteau. Boltzmann sampling of unlabelled structures. *Proceedings of the 9th Workshop on Algorithm Engineering and Experiments and the 4th Workshop on Analytic Algorithms and Combinatorics (ANALCO)*, pages 201–211, 2007.

[FGM97] Philippe Flajolet, Xavier Gourdon, and Conrado Martínez. Patterns in random binary search trees. *Random Structures & Algorithms*, 11(3):223–244, 1997.

[FO90] Philippe Flajolet and Andrew Odlyzko. Singularity Analysis of Generating Functions. *SIAM Journal on Discrete Mathematics*, 3(2):216–240, 1990.

[FS06] Dominique Foata and Marcel-Paul Schützenberger. *Théorie géométrique des polynômes eulériens*, volume 138. Springer, 2006.

[FS09] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

[FSS90] Philippe Flajolet, Paolo Sipala, and Jean Marc Steyaert. Analytic variations on the common subexpression problem. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 443 LNCS, pages 220–234. Springer, New York, 1990.

[FZV94] Philippe Flajolet, Paul Zimmerman, and Bernard Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoretical Computer Science*, 132(1-2):1–35, 1994.

[Gal73] Francis Galton. Problem 4001. *Educational Times*, 1(17):4, 1873.

[Gar78] Martin Gardner. Bells-versatile numbers that can count partitions of a set, primes and even rhymes. *Scientific American*, 238(5):24—-30, 1978.

[Gen17] Antoine Genitrini. *Combinatoire énumérative et analytique en Logique Propositionnelle et en Théorie de la Concurrence: Vers une quantification de l'expressivité des modèles*. PhD thesis, Sorbonne Université, 2017.

[GGKW20] Antoine Genitrini, Bernhard Gittenberger, Manuel Kauers, and Michael Wallner. Asymptotic enumeration of compacted binary trees of bounded right height. *Journal of Combinatorial Theory. Series A*, 172:1–43, 2020.

[Gil59] Edgar Nelson Gilbert. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

[GJ04] Ian Goulden and David M. Jackson. Combinatorial Enumeration, 2004.

[GKP94] Ronald L Graham, Donald E Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994.

[GMS18] Zbigniew Gołębiewski, Abram Magner, and Wojciech Szpankowski. Entropy and optimal compression of some general plane trees. *ACM Transactions on Algorithms*, 15(1):1–23, 2018.

[Gre91] Daniel H Greene. Labelled Formal Languages and Their Uses., 1991.

[GvRB$^+$12] Dick Grune, Kees van Reeuwijk, Henri E. Bal, Ceriel J.H. Jacobs, and Koen Langendoen. *Modern Compiler Design*. Springer Science & Business Media, 2012.

[Hen91] Peter Henrici. *Applied and computational complex analysis. {V}ol. 2*. Wiley Classics Library. John Wiley & Sons, Inc., New York, 1991.

[HM04] Elliotte Harold and Scott Means. XML in a Nutshell, 2004.

[Hwa98] Hsien-Kuei Hwang. On Convergence Rates in the Central Limit Theorems for Combinatorial Structures. *European Journal of Combinatorics*, 19(3):329–343, 1998.

[Inc44] Edward Lindsay Ince. *Ordinary Differential Equations*. Dover Publications, New York, 1944.

[Jan10] Svante Janson. A divertent generating function that can be summed and analysed analytically. *Discret. Math. Theor. Comput. Sci.*, 12(2):1–22, 2010.

[JKP11] Svante Janson, Markus Kuba, and Alois Panholzer. Generalized Stirling permutations, families of increasing trees and urn models. *Journal of Combinatorial Theory. Series A*, 118(1):94–114, 2011.

[Kem93] Rainer Kemp. Monotonically labelled ordered trees and multidimensional binary trees. In *International Symposium on Fundamentals of Computation Theory (FCT)*, pages 329–341. Springer, 1993.

[Kir84] Peter Kirschenhofer. On the average shape of monotonically labelled tree structures. *Discrete Applied Mathematics*, 7(2):161–181, 1984.

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

[Knu05] Donald E Knuth. *The Art of Computer Programming, Volume 4, Fascicle 3: Generating All Combinations and Partitions*. Addison-Wesley Professional, 2005.

[Knu13] Donald E Knuth. Two thousand years of combinatorics. *Combinatorics: Ancient & Modern*, pages 3–37, 2013.

[KP07] Markus Kuba and Alois Panholzer. On the degree distribution of the nodes in increasing trees. *Journal of Combinatorial Theory, Series A*, 114(4):597–618, 2007.

[KP16] Markus Kuba and Alois Panholzer. Combinatorial families of multilabelled increasing trees and hook-length formulas. *Discrete Mathematics*, 339(1):227–254, 2016.

[Mah92] Hosam M Mahmoud. *Evolution of random search trees*. Wiley New York, 1992.

[Mil06] Peter David Miller. *Applied Asymptotic Analysis*. Graduate studies in mathematics. American Mathematical Society, 2006.

[MM78] Amram Meir and John W Moon. On the altitude of nodes in random trees. *Canad. J. Math.*, 30(5):997–1015, 1978.

[MM03] Conrado Martínez and Xavier Molinero. Generic algorithms for the generation of combinatorial objects. In *In International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 2747, pages 572–581. Springer Berlin Heidelberg, 2003.

[Mol05] Xavier Molinero. *Ordered Generation of Classes of Combinatorial Structures*. Thesis, Universitat Politècnica de Catalunya, 2005.

[Moo74] John W Moon. *The distance between nodes in recursive trees*, pages 125–132. London Mathematical Society Lecture Note Series. Cambridge University Press, 1974.

[Mor04] Katherine Morris. On Parameters in Monotonically Labelled Trees. In *Mathematics and Computer Science III*, pages 261–263. Birkhäuser Basel, 2004.

[MP05] Katherine Morris and Helmut Prodinger. Monotone tree structures revisited, 2005.

[MTS18] Abram Magner, Krzysztof Turowski, and Wojciech Szpankowski. Lossless Compression of Binary Trees With Correlated Vertex Names. *IEEE Transactions on Information Theory*, 64(9):6070–6080, 2018.

[Muc97] Steven S Muchnick. *Advanced compiler design and implementation*. Morgan Kaufmann, San Francisco, CA, 1997.

[NEV86] Jacques NEVEU. Arbres et processus de Galton-Watson. *Annales de l'I.H.P. Probabilités et statistiques*, 1986.

[NW75] Albert Nijenhuis and Herbert S Wilf. *Combinatorial algorithms*. Computer science and applied mathematics. Academic Press, New York, NY, 1975.

[Odl95] Andrew M Odlyzko. Asymptotic enumeration methods. *Handbook of Combinatorics*, 2:1063–1229, 1995.

[Pip10] Nicholas Pippenger. The Hypercube of Resistors, Asymptotic Expansions, and Preferential Arrangements. *Mathematics Magazine*, 83(5):331–346, 2010.

[Pit84] Boris Pittel. On growing random binary trees. *Journal of Mathematical Analysis and Applications*, 103(2):461–480, 1984.

[Pit94] Boris Pittel. Note on the heights of random recursive trees and random m□ary search trees. *Random Structures & Algorithms*, 5(2):337–347, 1994.

[Pól54] Geörge Pólya. *Mathematics and Plausible Reasoning: Induction and Analogy in Mathematics*. Princeton University Press, 1954.

[PP07] Alois Panholzer and Helmut Prodinger. Level of nodes in increasing trees revisited. *Random Structures and Algorithms*, 31(2):203–226, 2007.

[PU83] Helmut Prodinger and Friedrich J Urbanek. On monotone functions of tree structures. *Discrete Applied Mathematics*, 5(2):223–239, 1983.

[RK11] Robert Sedgewick and Kevin Wayne. *Algorithms 4th Edition*. Addison-Wesley, 2011.

[Rou12] Olivier Roussel. *Génération aléatoire de structures ordonnées par le modèle de Boltzmann*. PhD thesis, Paris 6, 2012.

[Rus03] Frank Ruskey. Combinatorial generation. 2003.

[RW15] Dimbinaina Ralaivaosaona and Stephan Wagner. Repeated fringe subtrees in random rooted trees. In *12th Workshop on Analytic Algorithmics and Combinatorics 2015, (ANALCO)*, pages 78–88. SIAM, Philadelphia, PA, 2015.

[Say12] Khalid Sayood. *Introduction to Data Compression*. Morgan kaufmann, 2012.

[Sch70] Ernst Schröder. Vier combinatorische Probleme. *Zeitschrift für Mathematik und Physik*, 15:361–376, 1870.

[SDH$^+$04] Charles Semple, Philip Daniel, Wim Hordijk, Roderic D.M. Page, and Mike Steel. Supertree algorithms for ancestral divergence dates and nested taxa. *Bioinformatics*, 2004.

[Slo06] N.~J.~A. Sloane. *The on-line Encyclopedia of Integer Sequences*. Published electronically at {\tt http://www.research.att.com/~njas/sequences/}, 2006.

[SM95] Robert T Smythe and Hosam M Mahmoud. A survey of recursive trees. *Theory of Probability and Mathematical Statistics*, pages 1–28, 1995.

[SM01] Mike Steel and Andy McKenzie. Properties of phylogenetic trees generated by yule-type speciation models. *Mathematical Biosciences*, 170(1):91–112, 2001.

[Sta97] Richard P. Stanley. Hipparchus, Plutarch, Schröder, and Hough. *The American Mathematical Monthly*, 104(4):344–350, 1997.

[Ste16] Mike Steel. *Phylogeny Discrete and Random Processes in Evolution*, volume 89 of *CBMS-NSF regional conference series in applied mathematics*. SIAM, 2016.

[TT71] S. R. Tims and John Alfred Tyrrell. 3295. Approximate Evaluation of Euler's Constant. *The Mathematical Gazette*, 55(391):65, 1971.

[VC95] Daniel J. Velleman and Gregory S. Call. Permutations and Combination Locks. *Mathematics Magazine*, 68(4):243–253, 1995.

[Was87] Wolfgang Wasow. *Asymptotic expansions for ordinary differential equations*. Dover Publications, Inc., New York, 1987.

[WdB60] Edward Maitland Wright and Nicolaas Govert de Brujin. Asymptotic Methods in Analysis. *The Mathematical Gazette*, 44(348):144, 1960.

[Wil05] Herbert S Wilf. *Generatingfunctionology*. CRC press, 2005.

[YKKN07] Katsuhisa Yamanaka, Shin Ichiro Kawano, Yosuke Kikuchi, and Shin Ichi Nakano. Constant time generation of integer partitions. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 2007.

[ZYK13] Jie Zhang, En Hui Yang, and John C. Kieffer. Redundancy analysis in lossless compression of a binary tree via its minimal DAG representation. *IEEE International Symposium on Information Theory - Proceedings (ISIT)*, pages 1914–1918, 2013.

7.5 Appendix A

7.5.1 Proofs for the bijections presented in Section 7.3.1

Proof of Proposition 7.3.7. We proceed in two steps.

(1) Firstly, we'll show that for all $a \in \mathbf{A}_{n,\phi}$, $\mathcal{M}(a) \in V_{n,\phi}$.
- Let $v = \mathcal{M}(a)$, then $v_0 = sum(a) = n$,$v_1 = v_0 - a_0 = n - (n-k) = k$. Let us denote $m = max(a)$. We have that $v_{m+1} = 0$ and by definition $v_m = u_m > 0$ and so the subsequence of the last two elements is decreasing. But by the construction of v, we have that $\forall i, 1 \leq i \leq m-2, v_{m-i} = u_{m-i} + 2v_{m-i+1} - v_{m-i+2} \geq u_{m-i} + v_{m-i+1} \geq v_{m-i+1}$ and so the sequence is decreasing in its entirety.
- For all $0 \leq i \leq m$ we have that for the consecutive differences $s_i = v_i - vi + 1$ of elements in v, $s_m = v_m - v_{m+1} = u_m > 0$ and $s_{m-1} = v_{m-1} - v_m = u_{m-1} + 2v_m = u_{m+1} + 2s_m > s_m$. Once again by definition of v we have that $\forall i, 2 \leq i \leq m, s_{m-i} = v_{m-i} - vm - i + 1 = u_{m-i} + 3v_{m-i+1} - v_{m-i+2} = u_{m-i} + s_{m-i+1} + 2v_{m-i+1} > s_{m-i+1}$ since $v_{m-i+1} > 0$ in this interval.
- Finally, if $v_i - 2v_{i+1} + v_{i+2} > 0 \implies v_i > 2v_{i+1} - v_{i+2} \implies u_i > 0 \implies [z^{i+1}]\phi > 0$.

(2) We now show that M is indeed injective. Consider $a, b \in \mathbf{A}_{n,\phi}$ such that $a \neq b$. Then the fact that $M(a) \neq M(b)$ follows from the fact that the construction of M makes use of the occ function and the fact that $occ(a)$ cannot equal $occ(b)$ by definition of occ and $\mathbf{A}_{n,\phi}$.

□

Proof of Proposition 7.3.11. We show first that $\mathcal{G}(v) \in \mathbf{A}_{n,\phi}$ and then we show the injection.

(1) We show that for $v \in V_{n,\phi}$, $\mathcal{G}(v) \in \mathbf{A}_{n,\phi}$.
- Let $a = [a_0] + num(b,n) = \mathcal{G}(v)$, then $a_0 = v_0 - v_1 = n - k$. By definition of $num(b,n)$ the elements are placed in decreasing order. Let $l = last(v)$, $sum(a \backslash \{a_0\}) = \sum\limits_{i=1}^{l} ib_i = (l-1)v_{l-1} + \sum\limits_{i=1}^{l-2} ib_i = (l-1)v_{l-1} + \sum\limits_{i=1}^{l-2} i(v_i - 2v_{i+1} + v_{i+2}) = (l-1)v_{l-1} + \sum\limits_{i=1}^{l-2} iv_i - 2\sum\limits_{i=1}^{l-2} iv_{i+1} + \sum\limits_{i=1}^{l-2} iv_{i+2} = (l-1)v_{l-1} + \sum\limits_{i=1}^{l-2} iv_i - 2\sum\limits_{i=2}^{l-1}(i-1)v_i + \sum\limits_{i=3}^{l}(i-2)v_i = (l-1)v_{l-1} - (l-1)v_{l-1} + v_1 + 2v_2 - 4v_2 + 2\sum\limits_{i=2}^{l-1} v_i - 2\sum\limits_{i=3}^{l} v_i = v_1 - 2v_2 + 2v_2 = v_1 = k$.
- Since the spacings in v are deceasing all b_is are positive non zero($b_i = v_i - 2v_{i+1} + v_{i+2}$).
- If $1 \leq i \leq l$, $a_i > 0$, let $j = a_i$, then $j > 0 \implies b_j > 0 \implies v_j - 2v_{j+1} + v_{j+2} > 0 \implies [z^{j+1}]\phi > 0 \implies [z^{a_j+1}]\phi > 0$.

(2) For $v, w \in V_{n,\phi}$, such that $v \neq w$, then let $\mathcal{G}(v) = [v_0 - v_1] + num(t, n)$ and $\mathcal{G}(w) = [v_0 - v_1] + num(r, n)$ then $num(t, n) \neq numn(r, n)$. because at least one of $v_i \neq w_i$. Therefore $\mathcal{G}(v) \neq \mathcal{G}(w)$.

□

Proposition 7.3.12. Let $v = \mathcal{M}(a)$, $v_0 = sum(a)$ and $v_1 = v_0 - a_0$, and let $l = \mathcal{G}(\mathcal{M}(a)) = [l_0] + num(h, n)$, where h is an ordered list that result from the first part of $\mathcal{G}$ then $l_0 = v_0 - v_1 = sum(a) - sum(a) + a_0 = a_0$. Let $u = occ(a), m = max(a)$. We have that $v_{m+1} = 0, v_m = u_m$ we get that $l_m = a_m$. For all $1 \leq j \leq m$, $v_j = u_j + 2v_{j+1} - v_{j+2}$ and $h_j = v_j - 2v_{j+1} + v_{j+2} = u_j$. Then $u_j = h_j$. Since $u = h$ and $occ(a) = num(h, n)$ then $l_j = a_j$. □

Proof of Proposition 7.3.17. $\forall v = [v_0, \ldots, v_m] \in V_{n,\phi}$, $\mathcal{N}(v) \in P_{n,\phi}$. Since $v_0 = n, v_m = 0$, and the elements are strictly decreasing. The spacings make a partition of n. And if $p_i - p_{i+1} > 0 \implies v_i - 2v_{i+1} + v_{i+2} > 0 \implies [z^{i+1}]\phi > 0$. Now if $v \in V_{n,\phi}$ and $w \in V_{n,k}$ if $v \neq w$ then $\mathcal{N}(v) \neq \mathcal{N}(w)$ because the spacings between elements of v and w are necessarily decreasing(by definition). The spacings can not be equal everywhere. □

Proof of Proposition 7.3.18. Let $p = \mathcal{N}(v)$ and $w = \mathcal{F}(p)$. $w_0 = sum(p) = n$ then by induction, for $1 \leq i \leq m$, $w_i = w_{i-1} - pi - 1 = v_{i-1} - (v_{i-1} - v_i) = v_i$. □

n / k	8	9	10	11	12	13	14
9	$[9]$						
10	$[9, 1]$	$[10]$					
11	$[9, 1, 1]$	$[10, 1]$	$[11]$				
12	$[9, 1, 1, 1]$	$[10, 1, 1]$	$[11, 1]$	$[12]$			
13	$[4, 3, 2, 2, 2]$	$[10, 1, 1, 1]$	$[11, 1, 1]$	$[12, 1]$	$[13]$		
14	$[4, 2, 2, 2, 2, 2]$	$[10, 1, 1, 1, 1]$	$[11, 1, 1, 1]$	$[12, 1, 1]$	$[13, 1]$	$[14]$	
15	$[3, 2, 2, 2, 2, 2, 2]$	$[10, 1, 1, 1, 1, 1]$	$[11, 1, 1, 1, 1]$	$[12, 1, 1, 1]$	$[13, 1, 1]$	$[14, 1]$	$[15]$
16	$[2, 2, 2, 2, 2, 2, 2, 2]$	$[4, 2, 2, 2, 2, 2, 2]$	$[11, 1, 1, 1, 1, 1]$	$[12, 1, 1, 1, 1]$	$[13, 1, 1, 1]$	$[14, 1, 1]$	$[15, 1]$
17	$[2, 2, 2, 2, 2, 2, 2, 2, 1]$	$[3, 2, 2, 2, 2, 2, 2, 2]$	$[11, 1, 1, 1, 1, 1, 1]$	$[12, 1, 1, 1, 1, 1]$	$[13, 1, 1, 1, 1]$	$[14, 1, 1, 1]$	$[15, 1, 1]$
18	$[2, 2, 2, 2, 2, 2, 2, 2, 1, 1]$	$[2, 2, 2, 2, 2, 2, 2, 2, 2]$	$[11, 1, 1, 1, 1, 1, 1, 1]$	$[12, 1, 1, 1, 1, 1, 1]$	$[13, 1, 1, 1, 1, 1]$	$[14, 1, 1, 1, 1]$	$[15, 1, 1, 1]$
19	$[2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1]$	$[2, 2, 2, 2, 2, 2, 2, 2, 2, 1]$	$[11, 1, 1, 1, 1, 1, 1, 1, 1]$	$[12, 1, 1, 1, 1, 1, 1, 1]$	$[13, 1, 1, 1, 1, 1, 1]$	$[14, 1, 1, 1, 1, 1]$	$[15, 1, 1, 1, 1]$

Table 7.2: An example of Lemma 5.6.5, with $\phi_i = 2$ for $2 \leq i \leq 5$ and $\phi_i = (i-5)!$ for $i \geq 5$. We see that $n_0 = 18$ and $\eta = 10$.

Index

Index of Sequences

Index of Open questions