Enumerative perspectives on chord diagrams

by

Lukas Nabergall

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2022

© Lukas Nabergall 2022

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Sergi Elizalde Professor, Dept. of Mathematics, Dartmouth College
Supervisor:	Karen Yeats Associate Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal Member:	David Wagner Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal Member:	Kevin Purbhoo Associate Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal-External Member:	Jason Bell Professor, Dept. of Pure Mathematics, University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

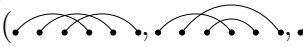
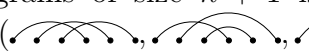
Statement of Contributions

Section 3.1 of this thesis is based on joint work with Ali Assem Mahmoud. The rest of the thesis is the sole work of Lukas Nabergall.

Abstract

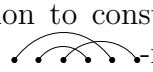
The topic of this thesis is enumerating certain classes of chord diagrams, perfect matchings of the interval $\{1, 2, \ldots, 2n\}$. We consider hereditary classes of chord diagrams that are restricted to satisfy one of several connectedness properties: connectivity, 1-terminality, and 1-sym-terminality (in order of increasing restrictedness). Such classes are defined by a set of minimal forbidden subdiagrams or patterns, and we focus on forbidding graphically-defined subdiagrams, in particular those whose intersection graph is isomorphic to a cycle. There are exactly two cycle diagrams of size n: the top cycle T_n and bottom cycle B_n. The class $\mathcal{D}(T_{>3})$ of diagrams avoiding a top cycle of size three or greater was previously shown to be equinumerous with the class of  -free diagrams by Jelínek, while the connected version of this class was put in bijection with planar bridgeless combinatorial maps by Courtiel, Yeats, and Zeilberger.

We begin by extending the recently developed automated enumeration framework Combinatorial Exploration of Albert, Baen, Claesson, Nadeau, Pantone, and Ulfarsson for enumerating combinatorial classes to chord diagrams. This framework algorithmically searches for a combinatorial specification for a given class by decomposing the class using a fixed set of decomposition strategies. Building off of their work, we construct a geometric version of chord diagrams amenable to Combinatorial Exploration and then describe a series of decomposition strategies for these geometric chord diagram classes. Most of these strategies are based on those developed for permutation classes by Albert, Baen, Claesson, Nadeau, Pantone, and Ulfarsson, but several appear to be new.

We then manually apply this framework to successfully enumerate a handful of diagram classes, including $\mathcal{C}($  $)$, $\mathcal{C}(T_{>3}, B_{>3})$, $\mathcal{D}($  $)$, $\mathcal{C}(B_{>3})$, and $\mathcal{T}(B_{>3})$. All but the second class have not previously been enumerated, and we give explicit closed-form formulas for each of them. As a corollary it follows that the number $|\mathcal{C}_{n+1}(B_{>3})|$ of bottom-cycle-free diagrams of size $n+1$ is equal to $|\mathcal{D}_n($  $)|$, while $|\mathcal{C}_{n+1}(T_{>3}, B_{>3})| = |\mathcal{D}_n($  $)|$. This appears to be a universal offset phenomenon—where connected classes are enumeratively equivalent to not-necessarily connected classes, the counting sequences are offset by 1. This points to a general map $\mathcal{C}_{n+1} \to \mathcal{D}_n$ restricting to bijections on these connected classes.

The restriction ψ of such a map can be explicitly obtained between 1-terminal diagrams $\mathcal{T}_{n+1}$ of size $n+1$ and diagrams $\mathcal{D}_n$ of size n, and we give a novel description ψ, prove that it is a bijection, and show that it restricts to a bijection between 1-terminal tree diagrams $\mathcal{T}(T_{>3}) = \mathcal{T}(T_{>3}, B_{>3})$ and noncrossing diagrams $\mathcal{D}($  $)$, thereby counting

the former. We then investigate the relationship between the map ψ and notion of higher terminality analogous to higher connectivity, as well as relate it to increasing trees and Stirling permutations. Finally, we obtain a characterization of closure under subdiagram avoidance for ψ and its inverse, giving bijections for an infinite set of pairs of restricted hereditary classes. We then obtain related results in a short study of 1-sym-terminal classes.

Diagram classes defined by forbidding top cycles require alternative methods to those used above. For this, we construct a novel tree-like decomposition for connected chord diagrams. This gives a recurrence relation for the number of connected diagrams counted by size and the index of the first so-called terminal chord in a total order known as the intersection order. Applying the decomposition to connected top-cycle-free diagrams gives a similar recurrence. We then use this decomposition to construct recursive bijections between between $\mathcal{C}(T_{>3})$ and the class of connected  -free diagrams, as well as triangulations of a disk. Via prior work of Brown, the latter leads to an explicit formula for the counting sequence of these diagram classes.

The recurrence relation for connected chord diagrams was previously implicitly obtained in work of Marie and Yeats giving chord diagram expansion solutions to certain Dyson-Schwinger equations from quantum field theory. Their proof was technically complex and passed to certain recursively-defined binary trees. We generalize this work using our connected diagram decomposition to solve a larger family of Dyson-Schwinger equations via weighted generating functions for weighted connected chord diagrams. We then discuss several conjectures towards obtaining similar solutions for more general and physically-realistic Dyson-Schwinger equations.

Acknowledgements

There's an old Polish proverb that says, "It's harder for the spider to catch the fly, than for the fly to catch the horse." Relatedly, it takes a village to write a PhD thesis. I'd like to take the time to acknowledge all the members of that wonderful village.

I owe much to my supervisor Karen Yeats. She provided a great deal of the foundation upon which this thesis rests, as well as steadfast guidance and the freedom to branch out into those bits of combinatorics that most interested me. I'd also like to thank the rest of the research group, especially Nick Olson-Harris, Ali Assem Mahmoud, William Dugan, and Iain Crump. Nick in particular for useful and fun discussions during the writing of the main paper this thesis is based on, and Ali for a fruitful collaboration and interesting recent conversations. May terminal chords live on forever.

While the math was excellent, the friends I met along the way were even better. In the antichain order, thanks to James Davies, Santiago Estupiñán, Soffía Árnadóttir, Valerie Gilchrist, Tina Chen, Andrew Jena, Weston Ford, Aristotelis Chaniotis, Shannon Veitch, Madison Van Dyk, Ben Moore, Martin Pei, Matthew Kroeker, Benjamin Anderson-Sackaney, Rose McCarty, Leah Cousins, Ali Assem Mahmoud, David Alemán Espinosa, Luke MacLean, Evelyne Esserre, Nathan Benedetto Proenca, Alan Wong, Fernanda Rivera Omana, Kazuhiro Nomoto, Spencer Wilson, Abbas Abou Daya, Nick Olson-Harris, Sean Kafer, Bethany Gunter, Tim Miller, Dmitry Sayutin, Ronen Wdowinski, Josh Gunter, Daniel Oliveira, Kelvin Chan, Iain Crump, Josephine Reynes, Hidde Koerts, Jason Qu, Naomi Graham, Sabrina Lato, Everett Patterson, Prerna Angrish, Evan Haithcock, and Mathieu Rundström. For the absurd conversations, the incomprehensible inside jokes, the rad parties, the strats, the late night escapades, and just being awesome.

Special thanks also go to friends gained away from Waterloo, whether in Florida or virtually. Firstly, Alejandro Navas, the best friend I could hope for throughout this journey. Mark Steele, who may have sent more links to random things of interest than most people experience in a lifetime. Micaela Newman, who made visits to Florida extra fun. And Eric Winsberg, Sudha Lakshmi, and Michael Alcorn, for bringing many doses of sanity and humor that were much needed amidst the historic and paradigm-shattering event that is the COVID-19 pandemic.

Further special thanks to my family, including my mom Jette, dad Scott, and brother Bjorn. Your unwavering support was vital, especially in the many calls enjoying so many excellent series together. Also, thanks to Oliver; even though you don't work your enthusiasm is unmatched.

Finally, I would also like to thank all members of the committee for reading this thesis, especially Kevin Purbhoo, who provided a most valuable critique, and I would like to thank the rest of the faculty and staff of the Combinatorics and Optimization department.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

A *combinatorial class*, or simply *class*, $\mathcal{A}$ is a countable set of objects equipped with a size function $|\cdot|: \mathcal{A} \to \mathbb{Z}_{>0}$ such that there are a finite number of objects $a \in \mathcal{A}$ of a given non-negative integer size $|a|$. Most unlabelled objects and objects labelled with a fixed set of labels studied in combinatorics form combinatorial classes; examples include permutations, graphs, integer partitions, set partitions, standard Young tableaux, words on a finite alphabet, and many others.

The most basic question one can ask about combinatorial classes is the enumeration problem: how many objects of a given size are there, that is, what is the cardinality of the finite set $\mathcal{A}_n$ of objects of size n in a class $\mathcal{A}$? An answer to this question ideally comes in the form of an explicit closed-form formula for the number $|\mathcal{A}_n|$, a closed-form expression for the ordinary or exponential generating functions $\sum_{a2A} x^{jaj}$ and $\sum_{a2A} \frac{x^{jaj}}{jaj!}$, or a size-preserving bijection from $\mathcal{A}$ to some other class $\mathcal{B}$ about which we have more information. Failing this, we may also obtain enumerative information in the form of e.g. a recurrence for the counting sequence $(|\mathcal{A}_n|)_{n2\mathbb{Z}_+}$, a recursion for a generating function of $\mathcal{A}$, or an asymptotic for $|\mathcal{A}_n|$. See the reference texts [95, 43, 5] for further details on this general context.

Solutions to the enumeration problem such as these can often be attained when the combinatorial class is structured in some sufficiently nice way. For many classes $\mathcal{A}$, the objects of $\mathcal{A}$ are equipped with a notion of subobjects; informally speaking, $a \in \mathcal{A}$ contains $b \in \mathcal{A}$ as a subobject if $|b|$ 6 $|a|$ and there is a way of identifying a 'part' of a with b. Typically, a and b are sets with some structure and identification occurs via a suitably defined homomorphism from b to a. Examples include patterns in permutations, subgraphs and minors of a graph, and subwords of a word. A class $\mathcal{A}$ is *hereditary* if it is closed under

taking subobjects, that is, for $a \in \mathcal{A}$ and a subobject b of a, we have $b \in \mathcal{A}$. Many of the most commonly studied classes in combinatorics are hereditary, including all of the above examples as well as, most notably, permutations avoiding a given set of patterns and minor-closed classes of graphs. Hereditary classes are uniquely defined by a set X of minimal subobjects not contained in any object of the class; we refer to these as the *minimal forbidden subobjects* of a hereditary class. Given a binary property $p : \mathcal{A} \to \{\texttt{True}, \texttt{False}\}$ on a class $\mathcal{A}$, we call a subclass $\mathcal{B}$ of $\mathcal{A}$ such that $p(b) = \texttt{True}$ for all $b \in \mathcal{B}$ a *restricted* class. Restricted hereditary classes defined by taking a restricted subclass of a hereditary class capture even more of the combinatorial classes studied in literature, including e.g. connected graphs, involutions, alternating Baxter permutations [27], set partitions with every block of a fixed size, etc.

In this thesis, we study the class $\mathcal{D}$ of chord diagrams. A chord diagram of size n is a perfect matching of the set $[2n] := \{1, 2, , 2n\}$ whose elements are called chords. Such objects have also been called matchings [58, 23], complete pairings [97], and interval systems [32], and can be viewed as set partitions with every block of size 2; we leave formal definitions and further details to Section 1.1 and later sections of this chapter. Here chord diagrams are rooted at the chord containing 1, but the term has also been used to refer to the unrooted object obtained after modding out by cyclic permutations of $[2n]$. For brevity, we will use the shortened term 'diagram' to refer to chord diagrams. The enumeration problem for $\mathcal{D}$ is easy to solve: by removing the root chord and considering all possible ways of reinserting it we can show that there are $(2n-1)!! = (2n-1)(2n-3)\cdots 3 \cdot 1$ diagrams of size n. Other objects counted by double factorials include double occurrence words, fixed-point-free involutions, ternary trees, increasing ordered trees, and Stirling permutations [50]. In the literature, chord diagrams seemed to have been first studied by Touchard [101]. Since then there has been much focus on the enumeration of various subclasses of chord diagrams by size as well as other statistics (e.g. [87, 97, 98, 81, 42, 58, 24, 85, 9, 30]).

The standard subobject notion for chord diagrams is that of a subdiagram. Roughly, a diagram D^0 is a subdiagram of a diagram D if D^0 can be obtained by removing a subset of the chords of D and standardizing the ground set of the resulting diagram to $[2|D^0|]$. We review the study of hereditary classes of chord diagrams with subdiagrams as the notion of subobject in Section 1.4, as well as related hereditary classes involving other types of objects. One of the most prominent and natural types of diagrams studied in the literature are connected diagrams, which are those diagrams C for which there is no proper interval of $[2n]$ that is the ground set of a subdiagram of C. Connectivity can also be defined via the intersection graph of a diagram C, the directed graph on the chords of C formed by adding an edge (c, c^0) if c^0 crosses c on the right; C is connected if and only if its intersection graph is weakly connected. Undirected intersection graphs of diagrams, known as circle

graphs, have been studied as pure graph objects (e.g. [80, 32]) and in particular play a major role in vertex minor theory [13]. We formally define and discuss connectivity for diagrams as well as related graphical features in Section 1.2.

Outside of enumerative combinatorics and graph theory, chord diagrams have also appeared in diverse areas such as knot theory [10], bioinformatics [55] and, most relevantly for the present purposes, physics. In 2013, Marie and Yeats [75] solved a certain Dyson-Schwinger equation from quantum field theory. Dyson-Schwinger equations are the equations of motion for quantum fields; see Chapter 4 and references mentioned therein for further details. The particular equation studied by Marie and Yeats has a recursive form similar to standard functional equations for the generating function of rooted trees. The solution of Marie and Yeats came as a series expansion indexed by connected diagrams and weighted by coefficients indexed by certain novel parameters of those diagrams. We define and further motivate the study of these parameters in Section 1.3. Hihn and Yeats [54] generalized their approach and obtained a chord diagram expansion solving a more general family of Dyson-Schwinger equations.

We will principally be interested in the enumeration and structure of restricted hereditary classes of diagrams defined by possibly imposing a connectedness property. Beyond connectivity itself, additional connectedness properties of interest are introduced in Section 1.3, partially motivated by the physics work introduced above.

After formally introducing the objects we will be working with, as well as all relevant properties, features, and basic enumerative and structural facts, we overview the principal body of literature on restricted hereditary classes and pattern avoidance in Section 1.4. We then summarize our results and contributions in Section 1.5, as well as give a roadmap of the organization of the thesis.

1.1 Set partitions and chord diagrams

We now turn to formally introducing the objects we will be working with. We begin in this section with set partitions, an important object across the field of combinatorics. Since we will not be dealing with integer partitions in this document, we use the generic term 'partition' to refer to set partitions. For further information than discussed here, Mansour [73] offers an extensive reference text on set partitions. We leave discussion on the enumeration of classes of partitions and diagrams to Section 1.4.

Definition 1.1.1. A *partition* P of a set X is a set of non-empty subsets of X such that each element of X is in exactly one subset in P. The elements of X are called *points*, the

subsets in P are called *blocks* or *parts*, and $|X|$ is the *size* of P. A point contained in a block of size one is called *isolated*. The partition P is *uniform* if every block of P has the same size and k-*uniform* if that size is k. A 2-uniform partition is also known as a *perfect matching*, while a partition with every block having size at most 2 is known as a *matching*.

When X is a set endowed with a total order then it is standard to index the blocks $B_1, \ldots, B_k$ of P according to the order of their smallest points, that is, $i < j$ if and only if $\min B_i < \min B_j$. In the sequel, unless otherwise stated, we assume all partitions are on totally ordered sets and index the blocks of a partition in this way, and we write $B_1 < \cdots < B_k$. For brevity we write blocks as strings of points and $P = B_1/B_2/\cdots/B_k$. We write $[x]_P$ for the block of P containing a point x. Every partition P of X can be identified with the equivalence relation $\sim_P$ such that for any $x, y \in X$ we have $x \sim_P y$ if and only if $[x]_P = [y]_P$. An example of a partition of the set $[7]$ is given below:

$$13/257/4/6$$

Partitions P are typically depicted visually by drawing the points of P abstractly as dots aligned horizontally and connecting consecutive points in a block by an overarching arc; e.g. the above example becomes

The arcs of a partition may cross (as the leftmost two do here) or nest. Such configurations were studied as an early example of pattern avoidance in set partitions [101, 87, 64, 23, 17, 95].

Definition 1.1.2. For a partition P, a set of four points $w < x < y < z$ not all in the same block forms a *crossing* if $w \sim_P y$ and $x \sim_P z$ and a *nesting* if $w \sim_P z$ and $x \sim_P y$. A k-*crossing* is a set of $2k$ points $x_1 < \cdots < x_k < y_1 < \cdots < y_k$ such that for any $1 \leq i < j \leq k$ the points x_i, x_j, y_i, y_j form a crossing. Similarly, a k-*nesting* is a set of $2k$ points $x_1 < \cdots < x_k < y_k < \cdots < y_1$ such that for any $1 \leq i < j \leq k$ the points x_i, x_j, y_j, y_i form a nesting. The partition P is k-*noncrossing* (resp. k-*nonnesting*) if it has no k-crossings (resp. no k-nestings); if $k = 2$ we simply refer to it as *noncrossing* or *nonnesting*, respectively.

Noncrossing partitions in particular appear in a surprisingly large number of contexts [76], including most prominently free probability theory [94].

A partition P of a set $X = \{x_1 < \ldots < x_n\}$ can also be represented by the word $w(P) = w_1 \cdots w_n$ with $w_i = j$ if $x_i \in B_j$. These words are called *restricted growth functions* because the growth of their letters from start to end is in a certain sense 'restricted'. In particular, they are defined by the conditions

1. $w_1 = 1$,

2. for $i > 2$, we have $w_i \leq 1 + \max_{j<i} w_j$.

Classes of restricted growth functions have been studied by Sagan [90] and Campbell, Dahlberg, Dorward, Gerhard, Grubb, Purcell, and Sagan [20]. There are two natural notions of containment of set partitions commonly considered in the literature.

Definition 1.1.3. A partition obtained by removing points from the blocks of P is a *subpartition* of P.[1] A *homomorphism* between two partitions P and Q of sets X and Y is a map $f\colon X \to Y$ such that $f(B)$ is a subset of a block of Q for all $B \in P$. It is an *isomorphism* if f has an inverse that is also a homomorphism. For totally ordered X and Y the map f is *order-preserving* if $f(x) < f(y)$ whenever $x < y$. The partition P is said to *contain* Q if there exists an order-preserving map f between Q and P that restricts to an isomorphism onto the image $f(Q)$, in which case $f(Q)$ is an *occurrence* of Q in P. The partition P is said to *r-contain* Q if P has a subpartition $P^0 = \{B_1^0, \ldots, B_\ell^0\}$ order-isomorphic to Q such that

$$B_1^0 < \cdots < B_\ell^0 \quad \text{and} \quad [\min B_1^0]_P < \cdots < [\min B_\ell^0]_P,$$

in which case P^0 is an r-occurrence of Q in P. The partition P *avoids* Q, or is Q-*free*, if it does not contain Q, and *r-avoids* Q if it does not r-contain Q.

A partition Q such that there exists a bijective order-preserving homomorphism $Q \to P$ is commonly known as a *refinement* of P. This notion generates a partial order on the set $\mathcal{P}_n$ of partitions of size n of $[n]$, making it not only a poset but a lattice (see e.g. [7] for further information).

Observe that a partition P is k-noncrossing if and only if it avoids $1(k+1)/2(k+2)/\cdots/k(2k)$ and k-nonnesting if and only if avoids $1(2k)/2(2k-1)/\cdots/k(k+1)$. Note that r-containment is stronger than containment, e.g. $145/23$ contains $12/34$ but does not r-contain it. The former containment notion is more naturally defined via restricted growth functions: P r-contains Q if and only if there is a subword of $w(P)$ order-isomorphic to $w(Q)$. This motivates generalizing containment of partitions to a containment relation between partitions and words.

Definition 1.1.4. Let u and w be words on totally ordered alphabets Σ_1 and Σ_2. A word obtained by removing letters from w is a *subword* of w. A *homomorphism* between

[1] Note that the ground set changes.

Figure 1.1: A chord diagram of size 6 with two indecomposable components.

$w = w_1 \cdots w_n$ and u is a map $f\colon \Sigma_2 \to \Sigma_1$ such that $f(w_1) \cdots f(w_n)$ is a subword of u. The word w *contains* u if there exists an order-preserving homomorphism f between u and w. Then a partition P *contains* the word u if $w(P)$ contains u.

This allows for a more general notion of a pattern in a partition, as we will use in Chapter 2.

We now focus on a special kind of partition, the main object of this document: chord diagrams.

Definition 1.1.5. A *chord diagram* D of size n is a perfect matching of $[2n] = \{1, 2, \ldots, 2n\}$. The blocks of D are referred to as *chords*. Chord diagrams are rooted objects with the chord containing 1 as the *root* of D. Note that the empty set $\emptyset$ is the diagram of size 0. By convention, the chords of D are written as ordered pairs (x, y) with $x < y$; x and y are called the *source* and *sink*, respectively, of chord (x, y).

Sources have also been referred to as first endpoints or openers, while sinks have correspondingly been called second endpoints or closers (e.g. [8]). Figure 1.1 displays a representative example of a chord diagram.

Definition 1.1.6. For a diagram D, a subset $C \subseteq D$ is a *subdiagram* of D.

Note that this is distinct from the notion of a subpartition of a diagram; in particular, there are such subpartitions that are not subdiagrams, namely those that contain isolated points. If a diagram D contains an occurrence C of diagram D' we often treat C and D' as interchangeable as appropriate. Observe that while containment and r-containment differ with a chord diagram only as the parent partition (e.g. 14/23 contains 12/3 but does not r-contain it), they agree when a chord diagram is both the parent and child.

Definition 1.1.7. Let D be a diagram. Two chords $c_1 = (x_1, y_1)$ and $c_2 = (x_2, y_2)$ of D with $x_1 < x_2$ are said to *cross* if $x_2 < y_1 < y_2$ and *nest* if $y_2 < y_1$. The pair (c_1, c_2) then forms a *crossing* or *nesting*, respectively. In the former case c_2 is a *right neighbor* of c_1 and c_1 is a *left neighbor* of c_2, while in the latter case c_1 is the *top chord* and c_2 is the *bottom chord* of the nesting. The *(directed) intersection graph* $G(D)$ of D has the chords as vertices and two chords c_1 and c_2 joined by a directed edge c_1c_2 if c_2 is a right neighbor of c_1. The *circle graph* of D is obtained by dropping directions from $G(D)$.

We extend the notion of nesting to subdiagrams in the obvious way: $C' \subseteq D$ is nested under $C \subseteq D$ if each chord of C' is nested under every chord of C. Note that the crossing and nesting notions on diagrams and partitions agree. Correspondingly, k-crossings, k-nestings, k-noncrossing, k-nonnesting, etc. also apply to diagrams; e.g. a diagram is called *nonnesting* (resp. *noncrossing*) if it contains no nestings (resp. no crossings).

There are several notions of reducibility for chord diagrams that are both useful for solving enumeration problems and interesting features in and of themselves (see e.g. [3] for further information on reducibility in a more general context). The most basic is decomposability.

Definition 1.1.8. Let D_1 and D_2 be diagrams. The *concatenation* of D_1 and D_2 is the diagram D_1D_2 of size $|D_1| + |D_2|$ that restricts to D_1 on the first $2|D_1|$ points and to D_2 on the remaining $2|D_2|$ points. A diagram D is *decomposable* if it can be expressed as the concatenation of two strictly smaller diagrams and *indecomposable* otherwise.

By iteration we can of course speak of concatenating more than two diagrams. Then the central fact is that chord diagrams can be uniquely decomposed into indecomposable pieces, matching the language.

Lemma 1.1.9. *Every diagram can be uniquely expressed as the concatenation of a sequence of indecomposable diagrams.*

Proof. For a diagram D, let D_1 be the maximal indecomposable subdiagram of D containing the root chord of D and $D' = D - D_1$. Inductively we get the unique decomposition $D_2 \cdots D_k$ of D' into indecomposable diagrams, so $D = D_1D_2 \cdots D_k$. This decomposition of D is clearly unique. □

The indecomposable subdiagrams provided by this lemma are referred to as the *indecomposable components* of the diagram, the maximal non-empty indecomposable subdiagrams.

1.2 Graphical features of diagrams

In this section we introduce a number of features and properties of chord diagrams motivated by graph theory.

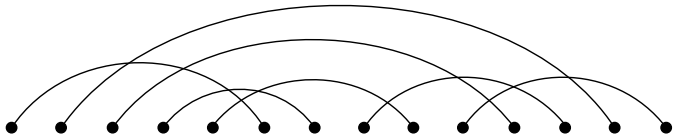


Figure 1.2: A connected diagram of size 7.

Definition 1.2.1. The *vertex connectivity* $\kappa(C)$, or simply *connectivity*, of a diagram C is the vertex connectivity $\kappa(G(C))$ of its circle graph, that is, one more than the maximum number of vertices whose removal cannot disconnect $G(C)$. The *edge connectivity* $\lambda(C)$ of C is defined analogously by removing edges instead of vertices. A diagram C is *k-connected* if $\kappa(C) > k$; in particular, it is *connected* if it is 1-connected. We consider the empty graph and, therefore, the empty diagram to be disconnected.

A disconnected diagram can be equivalently defined as one which can be partitioned into two nonempty subdiagrams with no crossings between them. If we also exclude nestings between the subdiagrams we get the stronger notion of decomposability from Definition 1.1.8. The diagram in Figure 1.1 is disconnected, while Figure 1.2 illustrates an example of a connected diagram. Connectivity was of primary interest from the beginning of the study of chord diagrams by Touchard [101] and Stein and Everett [97, 98]. There are two basic facts we will require about connected diagrams.

Lemma 1.2.2. *The diagram obtained by removing the root chord of a connected diagram can be uniquely expressed as a sequence of pairwise nested connected diagrams; in particular, it is indecomposable.*

Proof. For a connected diagram C, let C^0 be the diagram obtained by removing the root chord c of C. If C^0 cannot be expressed as a sequence of nested connected diagrams then there are maximal connected subdiagrams C_1 and C_2 of C^0 such that all the chords of C_1 lie to the left of all the chords of C_2. But then in C the root chord c cannot cross a chord of both C_1 and C_2, implying that C is disconnected, a contradiction. This gives the decomposition; uniqueness is apparent. □

Riordan [87] was the first to use this decomposition in the following useful form: every connected diagram C can be decomposed into the topmost connected subdiagram C_1 guaranteed by Lemma 1.2.2 and the connected subdiagram $C - C_1$ containing the root chord. This decomposition has been referred to as the *root-share decomposition* [75].

Lemma 1.2.3. *Every diagram can be uniquely decomposed into the following:*

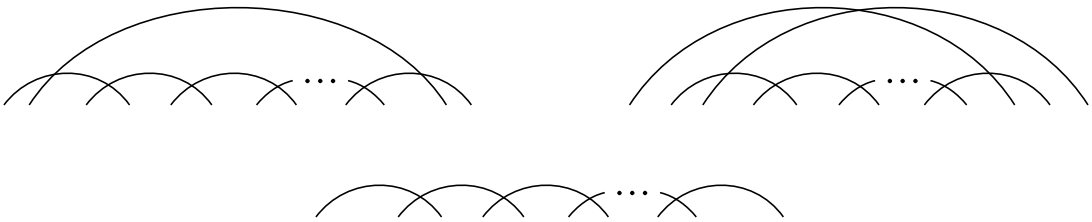


Figure 1.3: Top: top and bottom cycles, the only diagrams whose intersection graph form an induced cycle. Bottom: the unique diagram whose intersection graph forms a nonnesting induced path.

- *a noncrossing partition of* $[2n]$ *into even-sized blocks, and*
- *a sequence of connected diagrams.*

Proof. For a noncrossing partition P of $[2n]$ into even-sized blocks $B_1/B_2/\cdots/B_k$ and a sequence $(C_1, \ldots, C_k)$ of connected diagrams, clearly replacing each block B_i by perfect matching C_i on the ground set B_i gives a chord diagram. In the other direction, for a diagram D let $C_1, \ldots, C_k$ be the maximal connected subdiagrams of D ordered by their first sources. Then define B_i to be the ground set of C_i in D. Then clearly $B_1/B_2/\cdots/B_k$ is a noncrossing partition of $[2n]$ into even-sized blocks, as required. □

The connected diagrams returned by this lemma are the *connected components* of the diagram, the maximal non-empty connected subdiagrams.

There is an important theorem in graph theory characterizing k-connectivity in graphs in terms of the existence of certain subgraphs, induced paths. Recall that a subgraph H of a graph G is *induced* if H can be obtained from G by removing vertices of G.

Theorem 1.2.4 (Menger [77])**.** *A graph is k-connected if and only if it contains k internally-disjoint induced paths between any two vertices.*

This theorem applies to chord diagrams by way of their intersection graphs. While there are many diagrams of size n whose intersection graph is a path, the number of such diagrams reduces to one if we further require that the diagram be nonnesting; see Figure 1.3. There are many other types of graphs prominently studied in graph theory, especially as subgraphs; most notably among them, cycles. We also have a near representation uniqueness property for cycles in diagrams.

Lemma 1.2.5 (Bouchet [13])**.** *There are exactly two diagrams whose undirected intersection graph is isomorphic to a cycle of size n, the* top cycle *diagram T_n and* bottom cycle *diagram B_n.*

Figure 1.3 depicts these diagrams. They are named "top cycle" and "bottom cycle" because moving the top (i.e. rightmost) point of the top cycle to the bottom (i.e. root) of the diagram transforms it into a bottom cycle. Top cycle diagrams have appeared at least three times in the literature, including in a paper of Jelínek on certain hereditary classes of diagrams [58] and the work of Courtiel, Yeats, and Zeilberger [30] relating chord diagrams to certain combinatorial maps. In neither of these cases were they introduced from the graph-theoretic perspective.

We will often refer to chord diagrams by the graphical features of their intersection graphs, especially as it relates to excluding certain cycle diagrams. In this way we can speak of e.g. triangle-free diagrams, tree diagrams, bipartite diagrams, etc.

1.3 Ordered features of diagrams

We now turn to introducing order-based features of chord diagrams, some of which relate to the graphical features described previously. While the definitions in the previous sections are largely standard in the combinatorics literature, the following are more unique to our context and most of them first appeared in the paper of Marie and Yeats [75] (see also [54, 28, 30, 29]).

The directed intersection graph is acyclic, so it induces a partial order on the chords by reachability. We define several different total orders on the chords of a diagram D which extend this partial order.

Definition 1.3.1. The *standard order* $<$ of D is given by the order of the sources of the chords of D. For connected D, the *intersection order* $\prec$ is defined recursively as follows: starting with 1, label the root chord of C with the next available label, then remove the root and label the resulting connected components recursively in order of the standard order of their roots. The labels determine the intersection order.

The standard order will be used as the default order on a chord diagram unless stated otherwise. Note that the standard order and intersection order generally differ substantially (see Figure 1.4). If we replace 'connected' with 'indecomposable' and label the components in order of the reverse standard order of their roots then we get the *peeling order*, which

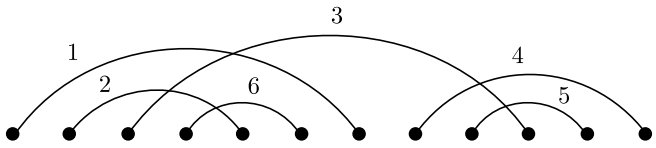


Figure 1.4: A connected diagram C with three terminal chords with indices 4, 5, and 6 in the intersection order, which differs from the standard order on C.

for some purposes is in fact equivalent to the intersection order (see [30] for details). We are interested in certain special chords which are the maximal elements in the reachability poset defined by the intersection graph.

Definition 1.3.2. Let D be a chord diagram. For $k \in \mathbb{N}$, a chord $c \in C$ is *k-terminal* if it is incident to at most $k-1$ outgoing edges in $G(D)$, that is, it has at most $k-1$ right neighbors. We refer to 1-terminal chords as simply *terminal*; these are the maximal elements in the reachability poset. We extend this language in an opposing direction to diagrams: D is *k-terminal* if there is no j-terminal chord before the jth-to-last[2] chord for all $1 \leq j \leq k$. Furthermore, the *terminality of* D is k if D is k-terminal but not $(k+1)$-terminal.

See Figure 1.4 for an example of terminal chords in a diagram and Figure 1.5 for an example of a 1-terminal diagram. Since the chord with rightmost sink in each connected component is necessarily terminal, it follows immediately from the definition that k-terminal diagrams are connected and the last $k+1$ chords form a clique; in particular, there are exactly j j-terminal chords for all $1 \leq j \leq k$. Furthermore, clearly k-terminal diagrams are also j-terminal for $1 \leq j < k$, while k-terminal chords are also ℓ-terminal for $\ell > k$. While, as indicated above, this makes k-terminality a kind of mirrored notion for diagrams relative to the notion for chords, the fact that restrictions on ($\leq k$)-terminal chords define k-terminality in diagrams motivates the shared terminology.

The indices of the terminal chords will play an important role in Chapters 3 and 4. Accordingly, we will denote the index of the jth terminal chord in the intersection order of a connected diagram C by $t_j(C)$. We now record a series of basic facts about these orders and 1-terminal objects, most of which have not appeared in the existing literature. For the rest of this section, let C be a connected diagram of size n with $t_1(C) = k$ and $c_1 \prec c_2 \prec \cdots \prec c_n$ be the chords of C in the intersection order. Note that if $t_1(C) = 1$,

[2]Referring, as always unless otherwise indicated, to the standard order.

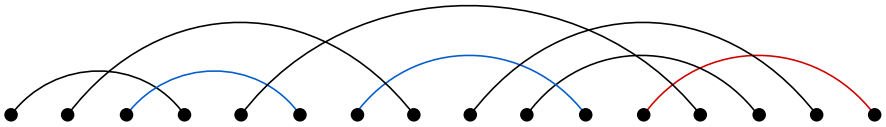


Figure 1.5: A 1-terminal diagram. The red chord is the unique terminal chord, and there is no nonnesting path between the two blue chords.

that is, the root chord is terminal, then it is the only chord of C, while if $t_1(C) = n$ then C is 1-terminal.

Lemma 1.3.3. *In the standard order, we have $c_1 < c_2 < \cdots < c_k$; that is, the intersection order and standard order agree in a relative sense up to the first terminal chord in the intersection order, which is the chord with rightmost sink, that is, containing $2n$.*

Proof. Note that the root chord comes first in the standard order and if we remove it to obtain an indecomposable diagram C', c_k is also the first terminal chord in the outermost connected component of C'. With these facts the result follows by applying induction to the outermost component. □

The second part of this lemma was observed by Courtiel and Yeats [28] in their work on terminal chords. As a consequence, the intersection order and standard order are equivalent on 1-terminal diagrams. The other direction does not hold; there are diagrams with multiple terminal chords in which the two orders agree (e.g. take a root crossing a set of terminal chords which all pairwise nest).

Lemma 1.3.4. *The indecomposable components $C_1, C_2, \ldots, C_m$ remaining after removing $c_1, c_2, \ldots, c_k$ have no right neighbors in C.*

Proof. Since the intersection order extends the partial order on chords induced by the directed intersection graph, the chords $c_1, \ldots, c_k$ can only cross chords in $C_1 \cup \cdots \cup C_m$ on the left. Since these chords are the only neighbors of $C_1, \ldots, C_m$ by specification, the statement follows. □

We have the following result, which leads to an analogue of Menger's theorem for 1-terminality.

Lemma 1.3.5. *There exists a nonnesting induced path in $\{c_1, c_2, \ldots, c_k\}$ from c_j to c_k for all $1 \leq j \leq k$. In particular, the chords $c_1, c_2, \ldots, c_k$ induce a 1-terminal subdiagram.*

Proof. If $n = 1$, the result holds trivially. Otherwise, remove the root and consider the outermost component D of the resulting indecomposable diagram. Inductively, for all $2 \leq j \leq k$ there is a nonnesting induced path $P_j \subseteq \{c_2, \ldots, c_k\}$ from c_j to c_k in D. Since C is connected, by the order extension property the source of c_2 lies before the sink of c_1. Furthermore, c_k either crosses the root or its source lies after the sink of the root. Thus some chord in P_2 crosses c_1; choose such a chord c_i with i maximum and let $P_2^{\emptyset}$ be the subpath of P_2 beginning at c_i. Then $\{c_1\} \cup P_2^{\emptyset}$ is a nonnesting induced path from c_1 to c_k in C. □

One can check that this proof actually implicitly constructs nonnesting induced paths to the first terminal chord defined by the property that the $(i+1)$th chord of the path is the rightmost neighbor in $\{c_1, c_2, \ldots, c_k\}$ of the ith chord. It also gives a characterization of 1-terminality.

Corollary 1.3.6. *The following are equivalent.*

(i) C *is 1-terminal.*

(ii) $C - c_1$ *is 1-terminal.*

(iii) There exists a nonnesting induced path from c *to* c_n *for all* $c \in C$.

Proof. Lemma 1.3.5 gives the equivalence of (i) and (iii), while the fact that the root c_1 is not a right neighbor of any chord straightforwardly implies that (i) and (ii) are equivalent. □

The above results begin to illustrate how 1-terminality can be thought of as a more well-behaved notion of 'ordered' connectivity. In particular, for vertex connectivity, a version of (ii) no longer applies and we drop "nonnesting" to get a version of (iii). We will see further evidence for this view later in the thesis. Note that top cycles are 1-terminal and bottom cycles are not. Figure 1.5 illustrates a 1-terminal diagram with two chords that are not mutually contained in a nonnesting induced path, indicating that Corollary 1.3.6 cannot be strengthened to get such a path between e.g. any two nonnesting chords absent a stronger hypothesis.

One can also symmetrize the notion of 1-terminal. Let D^R be the diagram obtained by reversing the order of the points of D. Define a diagram D to be *1-sym-terminal* if D and D^R are 1-terminal. A symmetric version of Corollary 1.3.6 holds for 1-sym-terminal diagrams.

Lemma 1.3.7. *The following are equivalent.*

(i) C *is 1-sym-terminal.*

(ii) There exists a nonnesting induced path from c *to both* c_1 *and* c_n *for all* $c \in C$.

1.4 Restricted hereditary classes and pattern avoidance

In this section we review much of the existing literature on pattern avoidance in permutations, set partitions, and chord diagrams. Pattern avoidance is the primary language used to frame and describe restricted hereditary classes in enumerative combinatorics. The literature on this subject is so vast that we will only touch on a small fraction of it. See the survey of Kitaev and Mansour [63] and reference text of Kitaev [62] for further information and references.

Recall that hereditary classes are uniquely defined by a set of minimal forbidden subobjects. For a class $\mathcal{A}$ equipped with a subobject notion, we write $\mathcal{A}(X)$ for the hereditary class with minimal forbidden subjects X.

1.4.1 Permutations

Let $\mathcal{S}$ be the class of permutations. As is the case for most of the commonly-studied classes in combinatorics, the enumeration problem for $\mathcal{S}$ is trivial: there are $n!$ permutations of size n, that is, $|\mathcal{S}_n| = n!$. In the enumerative context, the principal notion of subobject for permutations is that of a pattern. A permutation $\sigma \in \mathcal{S}_n$ has an *occurrence* of the *pattern* $\tau \in \mathcal{S}_k$ if there exist $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ such that there is an order-preserving isomorphism between $\sigma(i_1)\sigma(i_2)\cdots\sigma(i_k)$ and τ. Beyond $\mathcal{S}$ itself, there have been thousands of hereditary classes $\mathcal{S}(T)$ defined by avoiding a given set T of patterns enumerated over the last seventy years. We give a brief overview of some of the most prominent results.

A classic theorem of Erdős and Szekeres [40] states that any permutation of size $(r-1)(s-1)+1$ contains one of the patterns $12\cdots r$ or $s\cdots 21$. It follows that the number of permutations avoiding both of these patterns is finite for any given $r, s > 1$. Furthermore, it is trivial to see that $\mathcal{S}_r(21) = \{12\cdots r\}$ and $\mathcal{S}_r(12) = \{r\cdots 21\}$. Following Hammersley's

[53] enumeration of $\mathcal{S}(321)$, Knuth [67] showed that for any $\tau \in \mathcal{S}_3$, we have $|\mathcal{S}_n(\tau)| = C_n$, where

$$C_n = \frac{1}{n+1}\binom{2n}{n}$$

is the nth Catalan number (see [93, A000108]), which has the algebraic generating function $\frac{1-\sqrt{1-4x}}{2x}$. Sets of patterns that generate hereditary classes with the same counting sequence are typically called *Wilf-equivalent*, meaning that they are in the same Wilf-equivalence class,[3] so above result shows that the patterns of size three are all Wilf-equivalent. Simion and Schmidt [92] completed the enumeration of all permutation classes avoiding sets of patterns of length three, finding that there are ten Wilf-equivalence classes and they all have counting sequence formulas even simpler than than the Catalan numbers. Later, focus turned to enumerating classes $\mathcal{S}_n(\tau)$ with $\tau \in \mathcal{S}_4$. There are three Wilf-equivalence classes: $\{1234, 1243, 1432, 2143\}$, $\{1342, 2413\}$, and $\{1324\}$.[4] Gessel [51] obtained formulas for $|\mathcal{S}_n(1234)|$ and its nonalgebraic, holonomic generating function. Bona [11] then found formulas for $|\mathcal{S}_n(1342)|$ and its algebraic generating function. It remains a notoriously difficult open problem to enumerate the remaining Wilf-equivalence class, $\{1324\}$.

For going beyond these small number of classes excluding a single permutation of size 4, there has been much work on developing automatable methods for solving the enumeration problem that take advantage of the growing computation power available to researchers and the structured nature of hereditary permutation classes. Early algorithmic enumeration frameworks of this type include Zeilberger's [114] and Vatter's [104] finite enumeration schemes and Vatter's implementation of the regular insertion encoding [105]. Many of these methods are subsumed by the recently developed Combinatorial Exploration framework developed by Albert, Bean, Claesson, Nadeau, Pantone, and Ulfarsson [1]. There have been many frameworks for describing the structure of a combinatorial class developed in the literature, including the combinatorial specifications of Flajolet and Sedgewick [43], generating trees of Chung, Graham, Hoggatt, and Kleiman [25], combinatorial species of Joyal [61, 60], Bergeron, Labelle, and Leroux [4], and Pivoteau, Salvy, and Soria [86], and the ECO method of Barcucci, Lungo, Pergola, an Pinzani [2]. To be used, all of these require prior knowledge of the structure of a combinatorial class. In contrast, Combinatorial Exploration provides decomposition strategies that can be applied in an algorithmic fashion to automatically discover a structural description of a given combinatorial class, which can

[3]Note that we will occasionally use "class" to instead refer to equivalence classes of an equivalence relation.

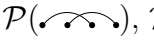
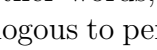
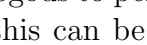
[4]This lists the permutations of $\mathcal{S}_4$ up to the reverse, complement, and inverse symmetries, which preserve Wilf-equivalence.

then be translated into an often solvable system of equations for the generating function of the input class. Albert, Bean, Claesson, Nadeau, Pantone, and Ulfarsson [1] applied Combinatorial Exploration to the enumeration of hereditary permutation classes. They were able to rediscover almost all known results from literature and either enumerate or obtain a combinatorial specification for numerous new classes. Most notably, they were able to enumerate all of the classes avoiding two or more distinct patterns of length four. We describe their method in more detail in Chapter 2, where we extend it to chord diagram classes.

1.4.2 Partitions and diagrams

Recall that $\mathcal{D}$ is the class of chord diagrams and write $\mathcal{P}$ for the class of set partitions. As previously mentioned, it can be easily shown that $|\mathcal{D}_n| = (2n-1)!!$ and the corresponding generating function is given by the differential equation $D(x) = 1 + xD(x) + 2x^2D'(x)$. The number of set partitions is well known to be given by the Bell numbers defined by the recurrence $b_n = \sum_{k=0}^{n} \binom{n}{k} b_k$ whose exponential generating function is given by $e^{e^x - 1}$. Note that $\mathcal{D}$ is not a hereditary subclass of $\mathcal{P}$.

Although not as thoroughly studied as permutation classes, there has been a large body of work on proper hereditary classes of partitions and diagrams. A *permutation diagram* is a diagram where all sinks lie to the right of all sources; the order σ of the sinks uniquely specifies a permutation diagram, which we denote by D_σ. Clearly permutation diagrams form the class $\mathcal{D}(\cap\ \cap)$ given by excluding two concatenated chords and $|\mathcal{D}_n(\cap\ \cap)| = n!$. Furthermore, if a hereditary class of either partitions or diagrams, respectively, does not exclude a permutation diagram then it necessarily contains at least $n!$ objects of size $2n$ or size n, respectively. Consequently, most focus has tended to remain on enumerating hereditary classes that do exclude at least one permutation diagram.

It is well known (see [95]) that the partitions and diagrams that are either noncrossing or nonnesting, that is, the classes $\mathcal{P}$(), $\mathcal{P}$(), $\mathcal{D}$(), and $\mathcal{D}$(), are counted by the Catalan numbers. In other words, such classes avoiding permutation diagrams of size two are enumeratively analogous to permutation classes avoiding patterns of size three. While for noncrossing classes this can be obtained using a simple decomposition by the root chord, for nonnesting classes it is generally necessary to pass through bijections to other Catalan objects or apply the well known correspondence involving standard Young tableaux and permutations of Robinson [88] and Schensted [91]. Gouyou-Beauschamps [52] studied the enumeration of  -free diagrams via involutions with no decreasing sequence of length 6, essentially giving a bijection between such diagrams and pairs of

noncrossing Dyck paths. Chen, Deng, Du, Stanley, and Yan [23] then extended these results to k-crossings and k-nestings of both diagrams and partitions, proving that their maximum size in a diagram or partition defines a pair of symmetrically distributed statistics and therefore $|\mathcal{D}_n(K_k)| = |\mathcal{D}_n(N_k)|$, where K_k is a diagram of k pairwise crossing chords and N_k is a diagram of k pairwise nesting chords. In particular, via a bijection between partitions and vacillating tableaux underlying their results they obtained the following.

Theorem 1.4.1 (Chen, Deng, Du, Stanley, and Yan [23, Corollary 12])**.** *The numbers of K_k-free and N_k-free diagrams of size n are equal to the number of closed lattice paths of length $2n$ in the set*

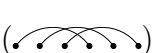
$$\{(a_1, a_2, \ldots, a_{k\ 1}) \mid a_1 > a_2 > \cdots > a_{k\ 1} > 0, a_i \in \mathbb{Z}\}$$

from the origin to itself with units steps in any coordinate direction or its negative.

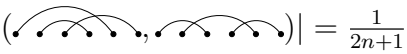
Corollary 1.4.2 (Gouyou-Beauchamps [52])**.** *There is a bijection between $\mathcal{D}_n(K_3)$ and pairs of noncrossing Dyck paths, so both have cardinality $C_nC_{n+2} - C_{n+1}^2$.*

We now record a number of other notable enumerative results obtained more recently, some of which closely relate to the classes we enumerate in this thesis.

Theorem 1.4.3. *The following hold.*

- *[58]* $|\mathcal{D}_n($  $)| = |\mathcal{D}_n($  $)| = |\mathcal{D}_n($  $)| = C_nC_{n+2} - C_{n+1}^2$.
- *[8]* The generating function for $\mathcal{D}($  $)$ is $\frac{54x}{1+36x\ \ (1\ \ 12x)^{3/2}}$.
- *[8] There are seven Wilf-equivalence classes for pairs of permutation diagrams of size 3, and we can obtain a functional equation or explicit formula for the generating function of all but $\{D_{321}, D_{231}\}$ and $\{D_{231}, D_{123}\}$.*
- *[21]* The generating function of the classes $\mathcal{D}($  $)$ and $\mathcal{D}($  $)$ *is*

  $$\frac{1}{1 - xC(x)C(C(x) - 1)},$$

  where $C(x)$ is the generating function of the Catalan numbers.
- *[21]* $|\mathcal{D}_n($  $)| = \frac{1}{2n+1}\binom{3n}{n}$.

1.5 Overview

In Chapter 2, we develop an extension of the automatable enumeration framework Combinatorial Exploration of Albert, Bean, Claesson, Nadeau, Pantone, and Ulfarsson [1] to chord diagrams. In Chapter 3, we apply this framework and other novel decomposition methods to enumerating a variety of restricted hereditary classes of chord diagrams. In Chapter 4, we describe an application of some of the results of Chapter 3 to Dyson-Schwinger equations from quantum field theory, generalizing prior work. In Chapter 5, we briefly discuss further conjectures and conclusions.

1.6 Notation

In this section, we list for quick reference important notation that will be used throughout the document.

C_n: Catalan numbers $\frac{1}{n+1}\binom{2n}{n}$

$t_1(C)$: index of the first terminal chord of a connected diagram C in the intersection order

1-terminal: a diagram with exactly one terminal chord

1-sym-terminal: a diagram D such that D and its reverse are 1-terminal

$\mathcal{D}_n(\mathcal{X})$: set of (chord) diagrams of size n with no subdiagram in $\mathcal{X}$

$\mathcal{C}_n(\mathcal{X})$: set of connected $\mathcal{X}$-free diagrams of size n

$\mathcal{T}_n(\mathcal{X})$: set of 1-terminal $\mathcal{X}$-free diagrams of size n

$\mathcal{ST}_n(\mathcal{X})$: set of 1-sym-terminal $\mathcal{X}$-free diagrams of size n

T_k, B_k: top cycle and bottom cycle diagrams, respectively, of size k; note that $T_3 = B_3 = K_3$

Chapter 2

A structural enumeration framework

We begin by extending the powerful enumeration framework of Albert, Bean, Claesson, Nadeau, Pantone, and Ulfarsson [1],[1] Combinatorial Exploration, to chord diagrams. Towards that aim we first formally define the framework.

Consider a combinatorial class $\mathcal{A}$. When successful, Combinatorial Exploration on $\mathcal{A}$ outputs a proof tree fully describing $\mathcal{A}$, that is, completely and uniquely specifying its counting sequence $(|\mathcal{A}_0|, |\mathcal{A}_1|, |\mathcal{A}_2|, \ldots)$. A *proof tree* for a class $\mathcal{A}$ is a rooted ordered tree whose vertices v index classes $\mathcal{B}^{(v)}$, where the root represents $\mathcal{A}$. Each vertex v with children $u_1, \ldots, u_m$ represents how the parent class $\mathcal{B}^{(v)}$ can be decomposed into the simpler child classes $\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)}$ or, conversely, how $\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)}$ can be combined to reconstruct $\mathcal{B}^{(v)}$. ABCNPU [1] formalized these decomposition relationships using so-called combinatorial strategies.

Definition 2.0.1. Let $\mathcal{Z}$ be the collection of all combinatorial classes and $m \in \mathbb{N}$. An m-ary *combinatorial strategy* S is a tuple $(d_S, r_S, (c_{S,(n)})_{n2\mathbb{N}})$ consisting of three components.

1. A *decomposition function* $d_S \colon \mathcal{Z} \to \mathcal{Z}^m \cup \{\texttt{DNA}\}$ that takes in a class $\mathcal{A}$ (the parent class) as input and outputs either an m-tuple of classes $(\mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(m)})$ (the child classes) or the symbol DNA; the latter, short for "does not apply", indicates that S *cannot be applied* to $\mathcal{A}$.

2. A *reliance pro le function* $r_S \colon \mathbb{N} \to \mathbb{Z}^m$ that outputs an m-tuple $(r_S^{(1)}(n), \ldots, r_S^{(m)}(n))$ of integers for each input natural number n.

[1]The group which hereafter will be referred to as ABCNPU.

3. An infinite sequence of *counting functions* $c_{S,(n)} \colon \mathbb{N}^{D_1} \times \cdots \times \mathbb{N}^{D_m} \to \mathbb{N}$ indexed by $n \in \mathbb{N}$ that outputs a natural number for every m-tuple of tuples of integers $w^{(1)}, \ldots, w^{(m)}$, where $D_i = \max\{0, r_S^{(i)}(n)\}$. The counting functions must have the property that if $d_S(\mathcal{A}) = (\mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(m)})$, then we have

$$c_{S,(n)}(w^{(1)}(n), \ldots, w^{(m)}(n)) = |\mathcal{A}_n|$$

for $w^{(i)}(n) = (|\mathcal{B}_0^{(i)}|, \ldots, |\mathcal{B}^{(i)}_{r_S^{(i)}(n)}|)$.

Combinatorial strategies exactly get at the notion of a class $\mathcal{A}$ decomposing into simpler classes $\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)}$: if strategy S applies to $\mathcal{A}$ and $d_S(\mathcal{A}) = (\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)})$, then using the reliance profile function and counting functions we can calculate the number of objects in $\mathcal{A}$ of size n from the number of objects in $\mathcal{B}^{(i)}$ of various sizes which are a function of n. In particular the calculation does not depend on the nature of the objects in the child classes, only their counts. This restricts the power of the Combinatorial Exploration framework by design, making it applicable to a broad universe of combinatorial classes.

As its name suggests, the reliance profile function encodes how objects in the parent class $\mathcal{A}$ rely on objects in the child classes $\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)}$. In particular, we say that $\mathcal{A}_n$ *relies on* $\mathcal{B}_j^{(i)}$ if $j \leq r_S^{(i)}(n)$, in which case the number of objects of size j in $\mathcal{B}^{(i)}$ is formally required to compute the number of objects of size n in $\mathcal{A}$.

When a strategy applies it produces a *combinatorial rule*

$$\mathcal{A} \xleftarrow{S} (\mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(m)}),$$

formally a tuple $(\mathcal{A}, (\mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(m)}), S)$. For example, consider the following strategy `Prod`:

- $d_{\texttt{Prod}}(\mathcal{A}) = (\mathcal{B}^{(1)}, \mathcal{B}^{(2)})$ if every object in $\mathcal{A}$ can be uniquely obtained as the concatenation of a pair of nonempty objects in $\mathcal{B}^{(1)} \times \mathcal{B}^{(2)}$, while $d_{\texttt{Prod}}(\mathcal{A}) = \texttt{DNA}$ otherwise,
- $r_{\texttt{Prod}}(n) = (n-1, n-1)$, and
- for all n,

$$c_{\texttt{Prod},(n)}((b_1^{(1)}, \ldots, b_{n-1}^{(1)}), (b_1^{(2)}, \ldots, b_{n-1}^{(2)})) = \sum_{i=1}^{n-1} |b_i^{(1)}||b_{n-i}^{(2)}|.$$

This corresponds to the usual product rule for generating functions [95], combinatorial classes [42], or species [5]. It assumes a suitably defined concatenation operation on

$\mathcal{B}^{(1)} \times \mathcal{B}^{(2)}$. This strategy applies to the class $\mathcal{I}^{(2)}$ of chord diagrams with at least two indecomposable components and yields, for example, the combinatorial rule

$$\mathcal{I}^{(2)} \xleftarrow{\texttt{Prod}} (\mathcal{I}, \mathcal{D}^{+}),$$

where $\mathcal{I}$ is the class of indecomposable diagrams and $\mathcal{D}^{+}$ is the class of nonempty diagrams. This follows by decomposing a diagram in $\mathcal{I}^{(2)}$ into the first indecomposable component (containing the root chord) and the rest of the diagram.

Observe that there is a special kind of combinatorial rule produced by a 0-ary strategy. Following ABCNPU [1], we call such a strategy a *verification strategy* because it indicates that the counting sequence of the input class has already been verified by an independent method. For each such class $\mathcal{A}$ we can define a verification strategy $V_{\mathcal{A}}$:

- $d_{V_{\mathcal{A}}}(\mathcal{A}) = ()$ and $d_{V_{\mathcal{A}}}(\mathcal{A}^{\emptyset}) = \texttt{DNA}$ for all $\mathcal{A}^{\emptyset} \neq \mathcal{A}$,
- $r_{V_{\mathcal{A}}}(n) = ()$, and
- for all n, $c_{V_{\mathcal{A}},(n)}$ is a 0-ary function that outputs $|\mathcal{A}_n|$.

Returning to proof trees, each parent-children pair of a proof tree corresponds to a combinatorial rule. In particular, each non-leaf vertex v with children $u_1, \ldots, u_m$ corresponds to a combinatorial rule

$$\mathcal{B}^{(v)} \xleftarrow{S} (\mathcal{B}^{(u_1)}, \ldots, \mathcal{B}^{(u_m)}),$$

while each leaf w corresponds to either a verification rule

$$\mathcal{B}^{(w)} \xleftarrow{V} ()$$

or the set $\mathcal{B}^{(w)}$ already appears as a non-leaf vertex in the tree. As ABCNPU [1] discuss, this makes proof trees largely equivalent to the *combinatorial specifications* developed by Flajolet and Sedgewick [43] in their symbolic combinatorial framework, with the combinatorial rules making up a proof tree forming as a set a combinatorial specification. See ABCNPU [1] for further discussion on the similarities and differences between the two notions, which can generally be used interchangeably. Note that a proof tree may contain rules produced by many different strategies, as is usually the case in practice.

Not all proof trees as we have defined them necessarily allow one to uniquely solve for the counting sequence of the root class. Consider possibly the simplest example combinatorial

specification

$$\begin{aligned} \mathcal{A} &\xleftarrow{S_1} \mathcal{B} \\ \mathcal{B} &\xleftarrow{S_1} \mathcal{A} \end{aligned} \tag{2.1}$$

where S_1 is a strategy identifying that $|\mathcal{A}_n| = |\mathcal{B}_n|$. Then this reduces to the system of recurrences

$$\begin{aligned} |\mathcal{A}_n| &= |\mathcal{B}_n| \\ |\mathcal{B}_n| &= |\mathcal{A}_n|, \end{aligned}$$

which of course yields no information about the counting sequences of either class $\mathcal{A}$ or $\mathcal{B}$. Generalizing this situation leads to the following definition.

Definition 2.0.2. A proof tree involving N combinatorial classes and its corresponding combinatorial specification are called ***productive*** if the infinite system of equations produced by the counting functions has a unique solution in $(\mathbb{C}^{\mathbb{N}})^N$. Otherwise the proof tree is ***trivial***.

Clearly productive proof trees are what we seek. The key discovery of ABCNPU [1] that enables Combinatorial Exploration to efficiently discover productive proof trees is that productivity can be ensured by a local property of a proof tree. In particular, as the above example suggests, placing certain restrictions on the allowed size and count relations generated by any strategy in the proof tree guarantees productivity.

Definition 2.0.3. An m-ary strategy S is ***productive*** if the following two conditions hold for all combinatorial rules

$$\mathcal{A} \xleftarrow{S} (\mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(m)})$$

and all $i \in [m]$.

1. For all $N \in \mathbb{N}$, if $\mathcal{A}_N$ relies on $\mathcal{B}_j^{(i)}$ then $j \leq N$.
2. If $\mathcal{A}_N$ relies on $\mathcal{B}_N^{(i)}$ for some $N \in \mathbb{N}$, then
   (a) $|\mathcal{A}_n| > |\mathcal{B}_n^{(i)}|$ for all $n \in \mathbb{N}$, and
   (b) $|\mathcal{A}_\ell| > |\mathcal{B}_\ell^{(i)}|$ for some $\ell \in \mathbb{N}$.

In a nutshell, an object in class $\mathcal{A}$ either relies only on strictly smaller objects, or it relies on objects of the same size, in which case the containing child class cannot be larger than $\mathcal{A}$ at any size and must be strictly smaller for at least one size. Enumeratively, these conditions match the informal notion of one class "decomposing" into "simpler" classes, and they are satisfied by many natural strategies.

Theorem 2.0.4 (ABCNPU [1]). *If a proof tree is composed entirely of rules derived from productive strategies then it is productive.*

While the strategy used in (2.1) fails Condition 2 of Definition 2.0.3 and therefore is not productive, ensuring it cannot on its own yield enumerative information, such *equivalence strategies* identifying two classes as equinumerous are still immensely useful for enumerating combinatorial classes. In particular they can be used to transform a class into an equivalent form which is structured in a way more amenable to decomposition via productive strategies. Formally, S is an equivalence strategy if $d_S(\mathcal{A}) = \mathcal{B}$ implies that $|\mathcal{A}_n| = |\mathcal{B}_n|$ for all n. The reliance profile function of an equivalence strategy can be assumed to be $r_S(n) = (n)$.

To deal with the fact that equivalence strategies are not productive ABCNPU [1] propose grouping combinatorial classes into equivalence classes. We say that class $\mathcal{A}$ is related to class $\mathcal{B}$ if $\mathcal{A} = \mathcal{B}$ or there is an equivalence strategy S that produces the rule $\mathcal{A} \overset{S}{\leftarrow} \mathcal{B}$ or the rule $\mathcal{B} \overset{S}{\leftarrow} \mathcal{A}$. Taking the transitive closure of this relation produces an equivalence relation R such that any two combinatorial classes in the same equivalence class are guaranteed to be equinumerous. We then generalize our notion of proof trees (or combinatorial specifications) to allow the vertices to represent these equivalence classes while combinatorial rules between parent-children pairs are reserved for those generated by productive strategies. We may identify two combinatorial classes as related during the construction of a proof tree in a way that enables us to terminate the search process much earlier than otherwise, leading to smaller proof trees and more success in finding them.

The process of Combinatorial Exploration involves searching for a productive proof tree for a given combinatorial class $\mathcal{A}$ by iteratively applying productive strategies and equivalence strategies from a pool of strategies. ABCNPU [1] describe efficient algorithms for executing this process on a computer and identifying when a productive proof tree has been obtained. It can of course also be executed by hand, potentially letting the user's intuition speed up the search process.

2.1 Gridding and tiling diagrams

It is not convenient to work directly with chord diagrams. There is in particular no obvious way to uniformly and compactly talk about the structure of the classes that arise when decomposing diagrams in certain natural ways. Building on previous work of Murphy and Vatter [78], ABCNPU [1] choose to solve this problem for permutations by endowing them with additional geometric structure called a *gridding*, leading to *gridded classes*. Such classes more naturally lead to effective decomposition strategies that are easier to describe, while preserving the content of the original objects. ABCNPU [1] also use this same geometric flavoring when informally discussing preliminary work on applying Combinatorial Exploration to the domains of alternating sign matrices, polynominoes, and set partitions. Since chord diagrams are a particular kind of set partition we base our definition of griddings for chord diagrams on the latter.

To produce more powerful and easily applicable decomposition strategies we take a general view on patterns in chord diagrams, wherein any word can be a pattern. Accordingly, we begin by gridding words. Henceforth assume, unless otherwise stated, that all words are on the alphabet $\mathbb{N}$ and all partitions of size n are partitions of the set $[n]$. Each word $w = w_1 \cdots w_n$ can be associated with the plot of the points $\{(w_i, i)\}_{i=1}^n$ in the Cartesian plane.

Definition 2.1.1. A *gridded word* of size n is a pair (w, E), where w is a word of length n called the *underlying word* and $E = (e_1, \ldots, e_n)$ for $e_i \in \mathbb{N}^2$ is the tuple of *positions*. The positions represent a placement of w onto $\mathbb{R}_+^2$ where the point corresponding to (w_i, i) with $e_i = (x, y)$ has been drawn in the square $[x, x+1) \times [y, y+1)$. The positions must be consistent with the word: for any direction $d \in \{\uparrow, \nearrow, \rightarrow, \searrow, \downarrow, \swarrow, \leftarrow, \nwarrow\}$, e_i is weakly in the dth direction relative to e_j if and only if (w_i, i) is in the dth direction relative to (w_j, j); e.g. for $d = \nearrow$,

$$x_i > x_j \text{ and } y_i > y_j \iff w_i > w_j \text{ and } i > j,$$

where $e_i = (x_i, y_i)$ and $e_j = (x_j, y_j)$.

See the next section for an example of a gridded word. When all the points of a gridded word have the same position e we simply write (w, e) in place of $(w, (e, \ldots, e))$. A square $[x, x+1) \times [y, y+1)$ and corresponding position (x, y) are both referred to as a *cell*. For a gridded word g and index i we write $g(i)$ for the point $((w_i, i), e_i)$ of g. Gridded words inherit the containment relation of their underlying words.

Definition 2.1.2. A gridded word $g = (w, (e_1, \ldots, e_n))$ *contains* another gridded word $h = (u, (f_1, \ldots, f_k))$ if w contains u and $f_j = e_{i_j}$ for all $1 \leq j \leq k$, where $i_1, \ldots, i_k$ are the indices of an occurrence of u in w; otherwise g *avoids* h. We call h a *gridded subword* of g and $(w_{i_1} \cdots w_{i_k}, (e_{i_1}, \ldots, e_{i_k}))$ an *occurrence* of h in g, and say that positions $e_{i_1}, \ldots, e_{i_k}$ *induce* h.

For clarity we typically refer to gridded words whose containment is being considered as *patterns*. We transfer griddings on words to griddings on partitions and chord diagrams via their restricted growth functions. All the associated language is carried over, while properties alter as appropriate; e.g. a *gridded (chord) diagram* of size n is a gridded word of size $2n$ such that the underlying word is the restricted growth function of a chord diagram, and a gridded subword of a gridded diagram that is itself a gridded diagram is referred to as a *gridded subdiagram*. Gridded objects, in particular gridded diagrams, inherit all previously defined properties of the underlying ungridded object. For example, a gridded diagram (D, E) is *connected* if D is connected. We further extend this notion and define gridded words in which some letter appears exactly once to be *disconnected*. This closes the connectedness notion on gridded diagrams under taking gridded subwords.

Definition 2.1.3. A gridded diagram G *avoids* a set of gridded patterns $\mathcal{O}$ if it avoids every gridded pattern in $\mathcal{O}$ and we define $\mathrm{Av}(\mathcal{O})$ to be the set of gridded diagrams avoiding $\mathcal{O}$. In contrast, the gridded diagram G *contains* a set of gridded patterns $\mathcal{R}$ if it does not avoid $\mathcal{R}$ and we define $\mathrm{Co}(\mathcal{R})$ to be the set of gridded diagrams containing $\mathcal{R}$.

Let $\mathcal{GW}$ be the set of all gridded words and $\mathcal{GW}^{(t,u)}$ be the set of gridded words whose positions lie within the rectangle $[0, t) \times [0, u)$. Furthermore, let $\mathcal{G}$ and $\mathcal{G}^{(t,u)}$ be the maximal subsets of $\mathcal{GW}$ and $\mathcal{GW}^{(t,u)}$, respectively, containing only gridded diagrams.

Definition 2.1.4. A *tiling* is a tuple $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$ where t, u are positive integers, $\mathcal{O}$ is a set of gridded patterns of $\mathcal{GW}^{(t,u)}$ called *obstructions*, $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_k\}$ is a set of sets of gridded patterns of $\mathcal{GW}^{(t,u)}$ called *requirements* with each set $\mathcal{R}_i$ referred to as a *requirement list*, and $\mathcal{L} = \{L_1, \ldots, L_m\}$ is a set of sets of points of $[0, t) \times [0, u)$ called *linkages*. A tiling $\mathcal{T}$ represents the set of gridded diagrams $G \in \mathcal{G}^{(t,u)}$ such that G avoids $\mathcal{O}$, G contains $\mathcal{R}_i$ for all $1 \leq i \leq k$, and the gridded subword of G induced by positions L_ℓ is connected for all $1 \leq \ell \leq m$. Denote this set by $\mathcal{G}(\mathcal{T})$.

Given a tiling $\mathcal{T}$, a cell e is called *empty* if it contains the obstruction of size 1, that is, there is an obstruction of $\mathcal{T}$ consisting of a single point placed in e. It is *nonempty* otherwise. A requirement in a requirement list of size 1 is called a *singleton requirement*.

Given any (ungridded) diagram class $\mathcal{D}(\mathcal{X})$, define the tiling $\mathcal{T}_{D(\mathcal{X})}$ to be the 1×1 tiling with no requirements or linkages and obstructions formed by gridding the diagrams in $\mathcal{X}$,

$$\mathcal{T}_{D(\mathcal{X})} = ((1,1), \{(M, (0,0)) : M \in \mathcal{X}\}, \emptyset, \emptyset).$$

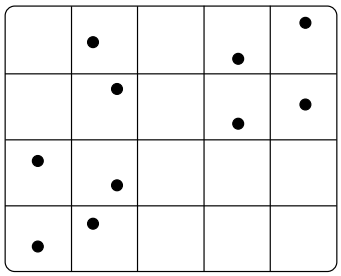
Similarly, for a connected diagram class $\mathcal{C}(\mathcal{X})$, define

$$\mathcal{T}_{C(\mathcal{X})} = ((1,1), \{(M, (0,0)) : M \in \mathcal{X}\}, \emptyset, \{\{(0,0)\}\}).$$

The map $D \to (D, (0,0))$ from $\mathcal{D}(\mathcal{X})$ ($\mathcal{C}(\mathcal{X})$ resp.) to $\mathcal{G}(\mathcal{T}_{D(\mathcal{X})})$ ($\mathcal{G}(\mathcal{T}_{C(\mathcal{X})})$ resp.) is a size-preserving bijection, so we can enumerate classes $\mathcal{D}(\mathcal{X})$ and $\mathcal{C}(\mathcal{X})$ by instead enumerating $\mathcal{T}_{D(\mathcal{X})}$ and $\mathcal{T}_{C(\mathcal{X})}$, respectively.

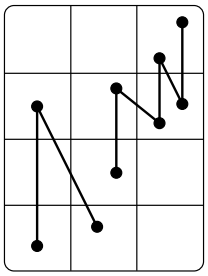
2.1.1 Visualizing gridded objects

Gridded words and tilings can be quite difficult to understand when written out, so it is useful to have a more digestible way to convey their content. For this we adapt the same type of graphical visualizations used by ABCNPU [1]. The entries of a gridded word are drawn as points on a rectangular grid, with each point placed in the cell indicated by its corresponding position. For example, the following depicts the gridded word $(1231453425, ((0,0),(1,0),(1,1),(0,1),(3,2),(4,2),(1,2),(3,3),(1,3),(4,3)))$, which also happens to be a gridded diagram:



When multiple gridded words are drawn on a single grid, as is often the case with tilings, we draw lines connecting points of a gridded word with the same letter (that is, same x coordinate of their position) and lines connecting the highest point of a given letter to the lowest point of the next letter. This makes it clear which points belong to which gridded words, in particular for gridded patterns in a tiling. While we avoid aligning points of separate gridded patterns horizontally or vertically to prevent confusion, their precise

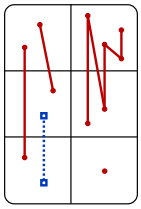
geometric layout relative to one another should not be regarded as significant. The following displays a representative example with the gridded words $(121, ((0,0),(1,0),(2,0)))$ and $(123123, ((1,1),(2,2),(2,2),(1,2),(2,3),(2,3)))$ depicted on a common grid.



For tilings, obstructions are drawn in red with solid lines and solid circular points, while requirements are drawn in blue with dotted lines and hollow square points. To accommodate many obstructions and requirements in a single grid or even a single cell, we use smaller points and thinner lines than depicted above. All requirements should be interpreted as singletons, that is, in a requirement list of size 1, unless otherwise specified. As an example, we depict the tiling $((2,3), \mathcal{O}, \mathcal{R}, \emptyset)$ with obstructions

$$\mathcal{O} = \{(11, ((0,0),(0,2))), (21, ((0,2),(0,1))), (1, (1,0)),\\ (123231, ((1,1),(1,1),(1,2),(1,2),(1,2),(1,2)))\}$$

and requirements $\mathcal{R} = \{\{(11, ((0,0),(0,1)))\}\}$:



We will also work with tilings with larger, in fact infinite, requirement lists and obstruction sets. This is difficult to depict visually in full granularity, so instead we use a more compact shorthand notation that loses no information, taking advantage of the fact that we will only be dealing with certain infinite sets of gridded patterns. Letting $\mathcal{G}_{\mathcal{X}}$ denote the set of gridded diagrams obtained by gridding a diagram in $\mathcal{X}$, there are seven such sets that will feature in our work:

- Gridded cycles $\mathcal{G}_{fT_{>3},B_{>3}g}$  ,

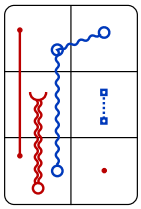
- Gridded top cycles $\mathcal{G}_{T_{>3}}$  ,
- Gridded bottom cycles $\mathcal{G}_{B_{>3}}$  ,
- Gridded diagrams formed by removing the root chord of a gridded top cycle $\mathcal{G}_{T^{\emptyset}_{>3}}$  ,[2]
- Gridded diagrams formed by removing the first chord of a gridded bottom cycle $\mathcal{G}_{B^{\emptyset}_{>3}}$  ,[3]
- Gridded diagrams formed by removing exactly one of the root chord or last chord of a gridded top cycle $\mathcal{G}_{fT^{\emptyset}_{>3},B^{\emptyset}_{>3}g}$  , and
- Gridded nonnesting paths $\mathcal{G}_{P_{>1}}$  ,

where $T^{\emptyset}_{>3}$ and $B^{\emptyset}_{>3}$ are the ungridded equivalents of $\mathcal{G}_{T^{\emptyset}_{>3}}$ and $\mathcal{G}_{B^{\emptyset}_{>3}}$ and we depict the line style used in our visualizations for each of the seven sets. We refer to the diagrams of these latter two ungridded sets as *partial top cycles* and *partial bottom cycles*, respectively. We specifically use subsets of these seven gridded sets where points may only lie in certain cells on a finite grid. For a given set of gridded words $\mathcal{H}$ and cells $e_1 < \ldots < e_k$ ordered lexicographically, define $\mathcal{H}_{e_1,\ldots,e_k}$ to be the subset of $\mathcal{H}$ containing all gridded words with points lying only in $e_1, \ldots, e_k$, the first point lying in e_1, and the last point lying in e_k. When $\mathcal{H}$ is an exceptional set $\mathcal{G}_{T^{\emptyset}_{>3}}$, $\mathcal{G}_{B^{\emptyset}_{>3}}$, and $\mathcal{G}_{fT^{\emptyset}_{>3},B^{\emptyset}_{>3}g}$ we further require that the second point of a gridded partial top cycle lies in e_1 and the second-to-last point of a gridded partial bottom cycle lies in e_k. When $\mathcal{H}$ is one of the seven sets defined above we visualize $\mathcal{H}_{e_1,\ldots,e_k}$ using the corresponding line style indicated in the above list. These are drawn in red and blue for requirements and obstructions, respectively. Intermediate cells $e_2, \ldots, e_{k-1}$ are indicated by hollow circles in each of those cells along the line.

For example, consider the following tiling $((2,3), \mathcal{O}^{\emptyset}, \mathcal{R}^{\emptyset}, \emptyset)$ with obstructions $\mathcal{O}^{\emptyset} = \{(\mathcal{G}_{B^{\emptyset}_{>3}})_{(0,0),(0,1)}, (11, ((0,0),(0,2)))\}$ and requirements $\mathcal{R}^{\emptyset} = \{(\mathcal{G}_{P_{>1}})_{(0,0),(0,2),(1,2)}, (11, (1,1))\}$. We include both finite and infinite patterns in this example for contrast.

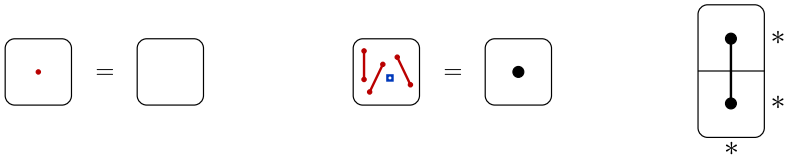
[2] Or, equivalently, removing the last chord of a bottom cycle.

[3] Equivalently, removing the last chord of a top cycle.



For linkages, we draw dashed boxes outlining adjacent cells in a linkage. Cells in the same linkage receive outlines of a common color.

Like ABCNPU [1], we apply additional shortcuts to commonly occurring features in order to simplify tiling visualizations. Empty cells are left blank, forgoing depicting the commonly occurring obstruction of size 1. This does not introduce ambiguity with cells devoid of obstructions and requirements because we will not be working with tilings that contain any such cells. Cells that must contain exactly one point are drawn with a single large black dot; this occurs when the cell contains the single point requirement and the obstructions 12, 21, and 11. If both endpoints of a chord are isolated to their own cells in this way, we join them by a black line to indicate the isolation of an entire chord. Furthermore, we add an asterisk at the bottom of each column containing exactly one chord and at the side of each row containing exactly one point. Here we demonstrate applying the shortcuts, as well as an example of a tiling consisting of a single chord isolated across two cells.



Combining all of the above visualization techniques, we display in Figure 2.1 the tiling $\mathcal{T} = ((3,4), \mathcal{O}, \mathcal{R}, \mathcal{L})$, where

$$\begin{aligned}
\mathcal{O} = \big\{&(1,(0,3)),(1,(1,0)),(1,(1,2)),(1,(2,0)),(1,(2,3)),(11,(1,1)),(12,(1,1)),\\
&(21,(1,1)),(11,(1,3)),(12,(1,3)),(21,(1,3)),(11,((0,0),(0,2))),(\mathcal{G}_{B^0_{>3}})_{(0,0),(0,1)},\\
&(21,((0,1),(0,2))),(123231,((2,1),(2,1),(2,2),(2,2),(2,2),(2,2)))\big\},\\
\mathcal{R} = \{&(1,(1,1)),(1,(1,3)),(\mathcal{G}_{P_{>1}})_{(0,0),(0,2),(2,2)},(11,(2,1))\}, \text{ and}\\
\mathcal{L} = \{&([0,1]\times[0,3]) - \{(2,0),(2,1),(2,2),(2,3)\}\}.
\end{aligned}$$

The gridded permutations in $\mathcal{G}(\mathcal{T})$ are those contained in $\mathcal{G}^{(3,4)}$ that

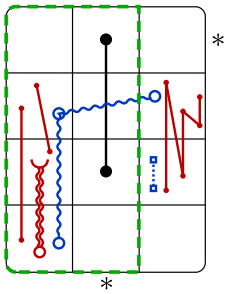


Figure 2.1: A tiling with obstructions, requirements, and linkages depicted using shortcuts.

- have no points in cells $(0,3)$, $(1,0)$, $(1,2)$, $(2,0)$, and $(2,3)$,
- have a single chord spanning cells $(1,1)$ and $(1,3)$ and no other points in those cells,
- have a nonnesting path that starts in cell $(0,0)$, intersects cell $(0,2)$, and ends in cell $(2,2)$,
- have a chord in cell $(2,1)$,
- do not have a chord spanning cells $(0,0)$ and $(0,2)$,
- do not have a gridded partial bottom cycle spanning cells $(0,0)$ and $(0,1)$,
- avoid the pattern 21 with the first point in cell $(0,2)$ and the second point in cell $(0,1)$,
- avoid the pattern 123231 with the first two points in cell $(2,1)$ and the last four points in cell $(2,2)$, and
- for which the gridded subdiagram induced by the first two columns is connected.

2.2 Combinatorial strategies

To enumerate the diagram classes $\mathcal{D}(\mathcal{X})$ and $\mathcal{C}(\mathcal{X})$ for a given $\mathcal{X}$, we work with the gridded equivalents $\mathcal{T}_{D(X)}$ and $\mathcal{T}_{C(X)}$ and apply strategies that decompose classes of the form $\mathcal{G}(\mathcal{T})$ for a tiling $\mathcal{T}$. We define these strategies in this section. While they actually act on gridded diagram classes, our definitions are all written in terms of the tilings themselves.

There are six main strategies: requirement insertion, obstruction/requirement/requirement list/linkage deletion, point/chord placement, row and column separation, obstruction/linkage inferral, and generalized factorization. While we view them each as single strategies they are all formally collections of strategies, with one strategy corresponding to each possible set of values of a group of parameters defining the strategy. All are equivalence strategies except the first and last, which are productive. Furthermore, all but linkage deletion, chord placement, linkage inferral, and generalized factorization were first defined in ABCNPU [1]. While these were only formally defined for permutations they are sufficiently close to identical to their versions for diagrams that we keep our descriptions of them brief and informal. The proofs that they are equivalence strategies or productive strategies also carry over. We formally define the remaining strategies that are specific to the chord diagram context over the next several subsections and provide appropriate proofs of their equivalency or productivity. Also, for each productive strategy, we give the corresponding equation for the generating functions $T(x)$ associated to every tiling $\mathcal{T}$ involved in a combinatorial rule produced by that strategy. For equivalence strategies these generating functions are equal.

For a set $\mathcal{Z}$ of gridded patterns, the Requirement Insertion strategy $\texttt{ReqIns}_{\mathcal{Z}}$ returns two tilings

$$\mathcal{T}_{\mathcal{O}} = ((t,u), \mathcal{O} \cup \mathcal{Z}, \mathcal{R}, \mathcal{L}), \qquad \mathcal{T}_{\mathcal{R}} = ((t,u), \mathcal{O}, \mathcal{R} \cup \{\mathcal{Z}\}, \mathcal{L})$$

for each applicable tiling $\mathcal{T} = ((t,u), \mathcal{O}, \mathcal{R}, \mathcal{L})$. This partitions $\mathcal{G}(\mathcal{T})$ into disjoint classes $\mathcal{G}(\mathcal{T}_{\mathcal{O}})$ and $\mathcal{G}(\mathcal{T}_{\mathcal{R}})$ containing those diagrams that avoid $\mathcal{Z}$ and those diagrams that contain $\mathcal{Z}$, respectively. Correspondingly $|\mathcal{G}(\mathcal{T})_n| = |\mathcal{G}(\mathcal{T}_{\mathcal{O}})_n| + |\mathcal{G}(\mathcal{T}_{\mathcal{R}})_n|$ for all n, so defining $d_{\texttt{ReqIns}_{\mathcal{Z}}}(\mathcal{T}) = \texttt{DNA}$ whenever $\mathcal{T}_{\mathcal{O}}$ or $\mathcal{T}_{\mathcal{R}}$ is empty makes Requirement Insertion a productive strategy. This is usually the workhorse starting strategy for decomposing any combinatorial class, whether diagrams or permutations—tiling $\mathcal{T}_{\mathcal{O}}$ is often substantially simpler to decompose or directly enumerate than $\mathcal{T}$ itself, while tiling $\mathcal{T}_{\mathcal{R}}$ introduces via the new requirement list $\mathcal{Z}$ some structure to further decompose on. The generating function equation corresponding to this strategy is

$$T(x) = T_{\mathcal{O}}(x) + T_{\mathcal{R}}(x).$$

The three simplification strategies Obstruction Deletion $\texttt{ObsDel}_M$, Requirement Deletion $\texttt{ReqDel}_{M,i}$, and Requirement List Deletion $\texttt{ReqListDel}_i$ each delete a condition whenever the condition is redundant, that is, deleting it leaves the set of gridded diagrams unchanged. In particular, for a tiling $\mathcal{T} = ((t,u), \mathcal{O}, \mathcal{R}, \mathcal{L})$:

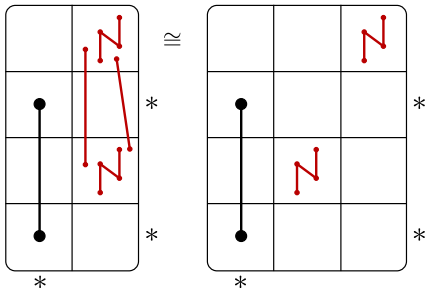


Figure 2.2: Column separation with $c = 1$ and $S = (1, 3)$ is used to isolate nonempty cells $(1, 1)$ and $(1, 3)$ from one another in a proof tree for noncrossing diagrams (see Section 2.3).

- for any gridded pattern g, $\texttt{ObsDel}_g$ returns the tiling $\mathcal{T}^\emptyset = ((t, u), \mathcal{O} - \{g\}, \mathcal{R}, \mathcal{L})$ if and only if $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^\emptyset)$.
- for any gridded pattern g and index i, $\texttt{ReqDel}_{g,i}$ returns the tiling

  $$\mathcal{T}^\emptyset = ((t, u), \mathcal{O}, \{\mathcal{R}_1, \ldots, \mathcal{R}_{i\ \ 1}, \mathcal{R}_i - \{g\}, \mathcal{R}_{i+1}, \ldots, \mathcal{R}_k\})$$

  if and only if $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^\emptyset)$.
- for any index i, $\texttt{ReqListDel}_i$ returns the tiling

  $$\mathcal{T}^\emptyset = ((t, u), \mathcal{O}, \{\mathcal{R}_1, \ldots, \mathcal{R}_{i\ \ 1}, \mathcal{R}_{i+1}, \ldots, \mathcal{R}_k\})$$

  if and only if $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^\emptyset)$.

Each of these are equivalence strategies by definition. ABCNPU [1] identify straightforward conditions in which each simplification strategy applies.

For a tiling $\mathcal{T}$, consider a row r and nonempty subset S of the nonempty cells in row r and let $S^\emptyset$ be the remaining nonempty cells in row r. The Row Separation strategy $\texttt{RowSep}_{r,S}$ returns a tiling $\mathcal{T}^\emptyset$ obtained from $\mathcal{T}$ by

- inserting a new row below row r,
- deleting all obstructions between cells in S and $S^\emptyset$, and
- moving the content of cells in S from row r to the new row while preserving column location.

In other words, Row Separation splits row r into two adjacent rows in a way that isolates cells in S from cells in $S^\emptyset$. This is possible if no gridded diagram in $\mathcal{G}(\mathcal{T})$ has a point in a cell in S whose value is larger than a point in a cell in $S^\emptyset$, which is guaranteed by the following condition: for all pairs of cells $(e, e^\emptyset) \in S \times S^\emptyset$, if e lies to the right of $e^\emptyset$ then $\mathcal{T}$ contains the obstruction $(12, (e^\emptyset, e))$, while if e lies to the left of $e^\emptyset$ then $\mathcal{T}$ contains the obstruction $(21, (e, e^\emptyset))$. The Column Separation strategy $\texttt{ColSep}_{c,S}$ is defined analogously for a column c and nonempty subset S of the nonempty cells in column c, with an analogous obstruction containment condition ensuring column separability. Figure 2.2 displays a simple example of applying the column separation strategy. ABCNPU [1] prove that both Row Separation and Column Separation are equivalence strategies.

For a gridded pattern g, the Obstruction Inferral strategy $\texttt{ObsInf}_g$ adds an obstruction g whenever the addition of g does not change the underlying set of gridded diagrams. In particular, for a tiling $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$, $\texttt{ObsInf}_g$ returns the tiling $\mathcal{T}^\emptyset = ((t, u), \mathcal{O} \cup \{g\}, \mathcal{R}, \mathcal{L})$ whenever $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^\emptyset)$. It is an equivalence strategy by definition. ABCNPU [1] describe two conditions, one more limited but computationally easy to verify and another fully general but computationally expensive, that largely transfer over to the diagram version. The one significant difference is that the restricted growth function property can also be used to infer obstructions.

It is also important to include verification strategies in our decomposition repertoire. From our previous definition of a verification strategy $V_\mathcal{A}$ for a class $\mathcal{A}$ we in particular get a verification strategy for each class $\mathcal{B} = \{G\}$ containing a single gridded diagram G, with associated generating function $B(x) = x^{|G|}$.

2.2.1 Linkage deletion

As with obstructions and requirements, we define a strategy Linkage Deletion that removes a linkage if its removal does not change the underlying set of gridded diagrams. For a set X of pairs of integers, define the equivalence strategy $\texttt{LinkDel}_X$ as follows.

- If $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$ is a tiling and $X \in \mathcal{L}$, define $\mathcal{T}^\emptyset = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L} - \{X\})$. If $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^\emptyset)$, then we define $d_{\texttt{LinkDel}_X}(\mathcal{T}) = \mathcal{T}^\emptyset$. Otherwise $d_{\texttt{LinkDel}_X}(\mathcal{T}) = \texttt{DNA}$.
- The reliance profile function is $r_{\texttt{LinkDel}_X}(n) = (n)$.
- The counting functions are $c_{\texttt{LinkDel}_X,(n)}((a_0, \ldots, a_n)) = a_n$.

Since it applies only when the underlying gridded diagrams do not change, Linkage Deletion is an equivalence strategy by definition. This leaves open the question of when

the strategy applies. While not obviously always determinable due to the inclusion of requirements, there are at least three easily detectable sufficient conditions for when a linkage can be deleted, the former two based on a distinct connectivity notion for the linkages themselves and the latter based on directly detecting a priori connectedness:

- Suppose a tiling $\mathcal{T}$ has distinct linkages $L_1, \ldots, L_i$ that induce a connected hypergraph and $L = \bigcup_{j=1}^{i} L_j$ is a linkage of $\mathcal{T}$. Then for every gridded diagram such that the gridded subdiagrams on position sets $L_1, \ldots, L_i$ are connected, the gridded subdiagram induced by positions L is also connected. So in this case we may apply the strategy $\texttt{LinkDel}_L$ to tiling $\mathcal{T}$.
- Suppose a tiling $\mathcal{T}$ has distinct linkages $L_1, \ldots, L_i$ such that each row intersects at most one linkage, the partition of the represented rows induced by the linkages $L_1, \ldots, L_i$ is connected, and $L = \bigcup_{j=1}^{i} L_j$ is a linkage of $\mathcal{T}$. Then we may similarly observe that connectedness on each of $L_1, \ldots, L_i$ guarantees connectedness on L, so in this case we may also apply the strategy $\texttt{LinkDel}_L$ to tiling $\mathcal{T}$.
- Suppose a tiling $\mathcal{T}$ has a linkage L such that amongst gridded diagrams in $\mathcal{G}(\mathcal{T}^{\emptyset})$ there are only finitely many gridded subwords induced by positions L and they are all connected. Then we may apply the strategy $\texttt{LinkDel}_L$ to $\mathcal{T}$.

Note that each of these conditions can be easily computed for all candidate linkages of a tiling.

2.2.2 Linkage inferral

We now define an inferral strategy for linkages that plays the same role as the inferral strategy for obstructions. In particular, it is often possible to add an additional linkage to a tiling without changing the underlying set of gridded diagrams. For a set of X of pairs of integers, the strategy $\texttt{LinkInf}_X$ is defined as follows.

- If $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$ is a tiling and $X \notin \mathcal{L}$, define $\mathcal{T}^{\emptyset} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L} \cup \{X\})$. If $\mathcal{G}(\mathcal{T}) = \mathcal{G}(\mathcal{T}^{\emptyset})$, then define $d_{\texttt{LinkInf}_X}(\mathcal{T}) = \mathcal{T}^{\emptyset}$. Otherwise $d_{\texttt{LinkInf}_X}(\mathcal{T}) = \texttt{DNA}$.
- The reliance profile function is $r_{\texttt{LinkInf}_X}(n) = (n)$.
- The counting functions are $c_{\texttt{LinkInf}_X,(n)}((a_0, \ldots, a_n)) = (a_n)$.

As with several previous strategies, Linkage Inferral does not change the underlying set of gridded diagrams, so it is by definition an equivalence strategy. To construct concrete computable conditions for when the strategy can be applied we make a simple observation: if a linkage can be deleted, it can also be added. In particular we can flip the role of $\mathcal{T}$ and $\mathcal{T}^\emptyset$ in each of the three linkage deletion conditions to obtain three cases when Linkage Inferral may be applied. Although this may seem odd, it is difficult to know beforehand whether it is more effective for finding a proof tree to include or not include a redundant linkage; including such a linkage could, for example, allow one to remove other linkages and thereby unlock an eventual proof tree.

2.2.3 Point and chord placement

Point placement is an equivalence strategy that acts on a tiling $\mathcal{T}$ by isolating one point of a singleton requirement in a cell of its own. Picking out an arbitrary instance of such a point usually does not reveal much useful structure, so we further specify a direction $d \in \{\downarrow, \uparrow, \leftarrow, \rightarrow\}$ in which the isolated point will be the most extreme instance of such a point in an occurrence of the given singleton requirement. In other words, the directions represent placing specifically the *bottommost*, *topmost*, *leftmost*, or *rightmost* instance of that point. There are two meaningful differences between point placement for gridded diagrams and point placement for gridded permutations: 1) for certain points, namely isolated points and sources, bottommost and leftmost (resp. topmost and rightmost) are equivalent, and 2) the column of the new cell is shared with one other point. Despite these differences the strategy is defined nearly identically so we will keep our description less than fully formal. For reference, see Figure 2.3 for a representative example.

Consider a tiling $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$ with $\mathcal{R} = \{\mathcal{R}_1, \ldots, \mathcal{R}_k\}$ and $\mathcal{L} = \{L_1, \ldots, L_m\}$. Suppose $\mathcal{T}$ contains a singleton requirement list $\mathcal{R}_1 = \{R\}$ and let $1 \leq \ell \leq |R|$ be an index and $d \in \{\downarrow, \uparrow, \leftarrow, \rightarrow\}$ be a direction. Write $e = (e_x, e_y)$ for the cell containing the placed point $R(\ell)$. Construct a new tiling $\mathcal{T}^\emptyset$ as follows. For a gridded word g, let $M_e(g)$ be the set of all possible gridded words obtained by splitting row e_y and column e_x into three rows and three columns, with cell e becoming nine cells and the points of g partitioned in all possible ways between the nine cells while maintaining their relative positions. Pictorially, $M_e(g)$ can be generated by simply adding two horizontal lines in row e_y and two vertical lines in column e_x in all possible locations with the constraint that the lines do not intersect any of the points of g. Let $M_e(\mathcal{S}) = \bigcup_{g \in \mathcal{S}} M_e(g)$ for a set $\mathcal{S}$ of gridded words. Furthermore,

define the map

$$\mu_e \colon (x,y) \mapsto \begin{cases} \{(x,y)\} & \text{if } x < e_x \text{ or } y < e_y, \\ \{(x,y+2)\} & \text{if } x < e_x \text{ or } y > e_y, \\ \{(x+i,x+j) : (i,j) \in [2]^2\} & \text{if } x = e_x \text{ and } y = e_y, \\ \{(x+2,y)\} & \text{if } x > e_x \text{ or } y < e_y, \\ \{(x+2,y+2)\} & \text{if } x > e_x \text{ or } y > e_y, \end{cases}$$

and set $\mu_e(A) = \bigcup_{a \in A} \mu_e(a)$. Then define $\mathcal{O}^{\emptyset} = M_e(\mathcal{O})$, $\mathcal{R}_i^{\emptyset} = M_e(\mathcal{R}_i)$ for $2 \leq i \leq k$, and $L_j^{\emptyset} = \mu_e(L_j)$ for $1 \leq j \leq m$. To ensure that cell (e_x+1, e_y+1) contains exactly one point, we add additional obstructions $A = \{(11,(e_x+1,e_y+1)),(12,(e_x+1,e_y+1)),(21,(e_x+1,e_y+1))\}$ and an additional requirement list $B = \{(1,(e_x+1,e_y+1))\}$. To ensure that no other cells in row e_y+1 have entries and only one other cell in column e_x+1 has entries, we add more obstructions

$$\begin{aligned} A_1 &= \{(1,(i,e_y+1)) : 0 \leq i < t, i \neq e_x+1\}, \\ A_2 &= \{(11,(e_x+1,j)) : 0 \leq j < u, j \neq e_y+1\}, \\ A_2^{\emptyset} &= \{(12,(e_x+1,j)) : 0 \leq j < u, j \neq e_y+1\}, \text{ and} \\ A_2^{\emptyset\emptyset} &= \{(21,(e_x+1,j)) : 0 \leq j < u, j \neq e_y+1\}. \end{aligned}$$

To guarantee that the isolated point in cell (e_x+1, e_y+1) is the ℓ'th point in an occurrence of R, we add the requirement list $\mathcal{R}_1^{\emptyset}$ containing the unique gridded word of $M_e(R)$ isolating point $R(\ell)$. Finally we include new obstructions to ensure that the isolated point in any gridded gridded is the most extreme point in the d direction that plays the role of $R(\ell)$ in any occurrence of R. In particular, we add the set D of gridded words of $M_e(R)$ that isolate a point in cell (e_x+1, e_y+1) and have their ℓ'th points occurrence further in the d direction than cell e. Combining all of the above, we have

$$\mathcal{T}^{\emptyset} = ((t+2,u+2), \mathcal{O}^{\emptyset} \cup A \cup A_1 \cup A_2 \cup A_2^{\emptyset} \cup A_2^{\emptyset\emptyset} \cup D, \{\mathcal{R}_1^{\emptyset}, \ldots, \mathcal{R}_k^{\emptyset}, B\}, \{L_1^{\emptyset}, \ldots, L_m^{\emptyset}\}).$$

With this we now define the point placement strategy.

– If $\mathcal{T}$ contains a singleton requirement list $\mathcal{R}_1 = \{R\}$, ℓ is an index with $1 \leq \ell \leq |R|$, and $d \in \{\downarrow, \uparrow, \leftarrow, \rightarrow\}$ is a direction, then we define

$$d_{\mathtt{PointPl}_{R,\ell,d}}(\mathcal{T}) = \mathcal{T}^{\emptyset}.$$

Otherwise we set $d_{\mathtt{PointPl}_{R,\ell,d}}(\mathcal{T}) = \mathtt{DNA}$.

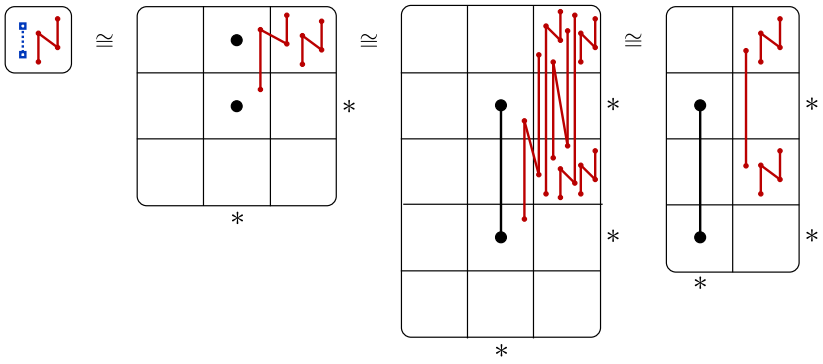


Figure 2.3: Starting with the leftmost tiling, we apply point placement with $R = (11, (0, 0))$, $\ell = 1$, and $d = \leftarrow$ to obtain the second tiling from the left. The third tiling is then obtained by further applying point placement to the second point of the requirement in cell $(1, 2)$. This sequence of point placements is equivalent to applying the chord placement strategy with $R = (11, (0, 0))$, $\ell_1 = 1$, $\ell_2 = 2$, and $d = \leftarrow$. Applying obstruction deletions and removing empty rows and columns then gives the rightmost tiling. This strategy chain occurs in the proof tree for noncrossing diagrams (see Section 2.3).

– The reliance profile function is $r_{\texttt{PointPl}_{R,\ell,d}}(n) = (n)$.

– The counting functions are $c_{\texttt{PointPl}_{R,\ell,d},(n)}((a_0, \ldots, a_n)) = a_n$.

The proof that point placement for gridded diagrams is an equivalence strategy is similar enough to the proof for gridded permutation point placement that we omit it (see [1]).

Theorem 2.2.1. *There is a size-preserving bijection between $\mathcal{G}(\mathcal{T})$ and $\mathcal{G}(\mathcal{T}^{\emptyset})$, implying that Point Placement is an equivalence strategy.*

Since every diagram is either empty or contains a chord it is often the case that we can use several point placements to isolate a designated chord in a class of gridded diagrams. Since this method will prove fruitful in finding proof trees, for compactness and ease of search we define a separate Chord Placement strategy as the composition of two point placements. In this case, by varying the order of placement, we get four truly distinct strategies, one for each coordinate direction: right and left place the chord source first, while up and down place the chord sink first. This matches the fact that the highest source in a tiling is in the rightmost chord.

– If $\mathcal{T}$ contains a singleton requirement list $\mathcal{R}_1 = \{R\}$, ℓ_1, ℓ_2 are indices with $1 \leq \ell_1 < \ell_2 \leq |R|$ such that $R(\ell_1)$ and $R(\ell_2)$ form a chord, and $d \in \{\downarrow, \uparrow, \leftarrow, \rightarrow\}$, then we define

$$d_{\texttt{ChordPl}_{R,\ell_1,\ell_2,d}}(\mathcal{T}) = \begin{cases} d_{\texttt{PointPl}_{R,\ell_2,d}}(d_{\texttt{PointPl}_{R,\ell_1,d}}(\mathcal{T})) & \text{for } d \in \{\leftarrow, \rightarrow\}, \\ d_{\texttt{PointPl}_{R,\ell_1,d}}(d_{\texttt{PointPl}_{R,\ell_2,d}}(\mathcal{T})) & \text{for } d \in \{\downarrow, \uparrow\}. \end{cases}$$

Otherwise $d_{\texttt{ChordPl}_{R,\ell_1,\ell_2,d}}(\mathcal{T}) = \texttt{DNA}$.

– The reliance profile function is $r_{\texttt{ChordPl}_{R,\ell_1,\ell_2,d}}(n) = (n)$.

– The counting functions are $c_{\texttt{ChordPl}_{R,\ell_1,\ell_2,d},(n)}((a_0, \ldots, a_n)) = a_n$.

Figure 2.3 depicts a simple example of chord placement.

2.2.4 Generalized factorization

Generalized factorization is a strategy that identifies when parts of a tiling can be split into several subtilings in a way that allows for recovering the original tiling. It straightforwardly generalizes the factorization strategy of ABCNPU [1] by allowing the subtilings to possibly share a common gridded subdiagram.

We say that two subsets S_1 and S_2 of cells of a tiling are *weakly non-interacting* if there is no obstruction, requirement list, or linkage involving cells in both S_1 and S_2. The subsets are further *(strongly) non-interacting* if they are weakly non-interacting and no cell of S_1 shares a row or column with any cell of S_2.

Consider a tiling $\mathcal{T} = ((t, u), \mathcal{O}, \mathcal{R}, \mathcal{L})$. A cover $Q = \{A_1, \ldots, A_\ell\}$ of the nonempty cells of $\mathcal{T}$ is *factorizable* if, for all $1 \leq i \leq \ell$, $A_i - \bigcup_{j \neq i} A_j$ is nonempty and non-interacting with all nonempty cells not in A_i, all pairs of nonempty sets $A_i \cap A_j$ and $A_i \cap A_h$ for $j, h \neq i$ are either equal or disjoint, and each such intersection set is weakly non-interacting with all other cells and strongly non-interacting with all parts of Q that do not contain it.

Suppose $\mathcal{T}$ factors in this way into two tilings $\mathcal{V}^{(1)}$ and $\mathcal{V}^{(2)}$ which intersect on the subtiling $\mathcal{U}$. The non-interactivity conditions imply that each gridded diagram G of size n that can be drawn on $\mathcal{T}$ can be formed uniquely from a pair H_1, H_2 where H_1 can be drawn on $\mathcal{V}^{(1)}$, H_2 can be drawn on $\mathcal{V}^{(2)}$, H_1 and H_2 share a common gridded subdiagram F that can be drawn on $\mathcal{U}$, and $|H_1| + |H_2| - |F| = n$. For fixed F, write $\mathcal{V}^{(1)}[F]$ and $\mathcal{V}^{(2)}[F]$ for the sets of all such H_1 and H_2, respectively. For our applications $\mathcal{U}$ should typically define a simpler set of gridded matchings than either $\mathcal{V}^{(1)}$ or $\mathcal{V}^{(2)}$, which will either themselves

pick out relatively simple classes or ones that have appeared before in a proof tree. So we may allow the cardinalities $|\mathcal{U}_i|$ in our data and aim to be able to compute $|\mathcal{T}_n|$ from $\{|\mathcal{V}_i^{(1)}|, |\mathcal{V}_i^{(2)}|, |\mathcal{U}_i|\}_{i=0}^n$. If $\mathcal{U}$ contains more than one gridded diagram we have

$$|\mathcal{T}_n| = \sum_{F \mathsf{2} \mathcal{U}} \sum_{i=jFj}^{n} |\mathcal{V}_i^{(1)}[F]||\mathcal{V}_{n\ i+jFj}^{(2)}[F]|,$$

and it is clearly not in general possible to remove the dependency on F. On the other hand, if $|\mathcal{U}| = 1$ then every gridded diagram in $\mathcal{V}^{(1)}$ and $\mathcal{V}^{(2)}$ contains the unique element F of $\mathcal{U}$ and the above expression simplifies to

$$\begin{aligned} |\mathcal{T}_n| &= \sum_{i=jFj}^{n} |\mathcal{V}_i^{(1)}||\mathcal{V}_{n\ i+jFj}^{(2)}| \\ &= \sum_{i=f}^{n} |\mathcal{V}_i^{(1)}||\mathcal{V}_{n\ i+f}^{(2)}|, \end{aligned}$$

where $f = \min\{j : |\mathcal{U}_j| = 1\} = |F|$, as desired. We could redo this example for the simpler case where $\mathcal{V}^{(1)}$ and $\mathcal{V}^{(2)}$ are disjoint and get the same result with $f = 0$. We now generalize this and formally define the factorization strategy, splitting a tiling along non-interacting pieces that pairwise intersect in at most a single fixed gridded diagram.

– Let Q be a factorizable cover of $\mathcal{T}$ with the parts $A_1, \ldots, A_\ell$ indexed in increasing order by their lexicographically smallest cell, with ties broken lexicographically. Write $\mathcal{V}^{(1)}, \ldots, \mathcal{V}^{(\ell)}$ for the subtilings induced by the parts of Q and, for all $1 \mathsf{6} i \neq j \mathsf{6} \ell$, write $\mathcal{U}^{(ij)}$ for the subtiling induced by the intersection $A_i \cap A_j$. Assume that each subtiling $\mathcal{V}^{(i)}$ contains at least one gridded diagram of size at least 1 and there is a gridded diagram in $\mathcal{V}^{(i)}$ that is not in $\mathcal{U}^{(ij)}$ for all j. Furthermore, assume that $|\mathcal{U}^{(ij)}| \in \{0, 1\}$ and let $(x_1, y_1), \ldots, (x_p, y_p)$ be the lexicographically smallest cells that respectively intersect each of the unique gridded diagrams in a nonempty subtiling $\mathcal{U}^{(ij)}$. Let $n_1, \ldots, n_p$ be the sizes of these gridded diagrams and define $I_q = \{i : (x_q, y_q) \in A_i\}$ for all $1 \mathsf{6} q \mathsf{6} p$. Define

$$\begin{aligned} N &= \left(\sum\{n_q : i \in I_q \text{ and } i \neq \min I_q\}\right)_{i=1}^{\ell}, \\ S &= \{i \in [\ell] : |\mathcal{V}_{N_j}^{(j)}| = 0 \text{ for some } j \neq i\}, \end{aligned}$$

and

$$d_{\mathsf{Factor}_{Q,N,S}}(\mathcal{T}) = (\mathcal{V}^{(1)}, \ldots, \mathcal{V}^{(\ell)}).$$

If either Q, N, or S is incompatible or one of the above assumptions is violated, we set

$$d_{\mathtt{Factor}_{Q,N,S}}(\mathcal{T}) = \mathtt{DNA}.$$

– The reliance profile function is

$$r_{\mathtt{Factor}_{Q,N,S}}(n) = (r^{(1)}(n), \ldots, r^{(\ell)}(n)),$$

where

$$r^{(i)}(n) = \begin{cases} n-1 & \text{if } i \in S, \\ n & \text{else.} \end{cases}$$

– Writing

$$v^{(i)} = \begin{cases} (v_0^{(i)}, \ldots, v_{n\ 1}^{(i)}) & \text{if } i \in S, \\ (v_0^{(i)}, \ldots, v_n^{(i)}) & \text{else,} \end{cases}$$

and

$$I = \{(i_1, \ldots, i_\ell) \in \prod_{i=1}^{\ell} [N_i, n + N_i] : i_1 - N_1 + \cdots + i_\ell - N_\ell = n \text{ and } i_j \neq n \text{ if } j \in S\},$$

the counting functions are

$$c_{\mathtt{Factor}_{Q,N,S},(n)}(v^{(1)}, \ldots, v^{(\ell)}) = \sum_{(i_1,\ldots,i_\ell) 2 I} v_{i_1}^{(1)} \cdots v_{i_1}^{(1)}.$$

The generating function equation corresponding to the strategy $\mathtt{Factor}_{Q,N,S}$ is

$$T(x) = \prod_{i=1}^{\ell} \frac{V^{(i)}(x)}{x^{N_i}}.$$

The key properties are the following, the proofs of which are sufficiently similar to those for the factorization strategy in [1] that we omit them.

Lemma 2.2.2. *Factor is a combinatorial strategy.*

Theorem 2.2.3. *The Factor strategy is productive.*

2.2.5 Additional strategies

We will require several other equivalence strategies that we briefly describe here. See [1] and the associated PermPal database for related strategies.

Requirement Inferral is an equivalence strategy similar to Obstruction Inferral that adds a requirement whenever the requirement is redundant. Cell Shuffling is an equivalence strategy that moves a set of cells (and all requirements, obstructions, and linkages) across a tiling whenever it does not change the counting sequence of the underlying set of gridded diagrams. Cell Splitting is an equivalence strategy that splits a cell in two and moves all cells above and to the right up and to the right by one cell. This is applicable whenever there is a gridded nonnesting paths $\mathcal{G}_{P_{>1}}$ obstruction with at least one point in the cell and such a point is not the first or last point of the path obstruction.

2.3 Example applications

We now perform a preliminary demonstration applying this framework to enumerate two simple diagram classes, $\mathcal{D}($  $)$ and $\mathcal{C}($$,$$)$. Both of these classes are counted by the Catalan numbers, which we prove here using the Combinatorial Exploration framework. While this is a classical result for the former [95], as far as we know neither the latter nor a verification of its counting sequence has appeared previously in the literature. We give a proof tree for each class, describe every combinatorial rule involved in the tree, and then derive the generating function for the Catalan numbers from the resulting system of equations. Since verifying that a proof tree is correct simply involves a (usually lengthy) calculation ensuring that each strategy was applied correctly and that the tree is indeed a valid proof tree, in the sequel we omit the extended description of each rule.

Figure 2.4 exhibits a proof tree for $\mathcal{D}($$)$. It closely tracks the proof tree for noncrossing set partitions given by ABCNPU [1] and formalizes the standard decomposition for noncrossing diagrams. The root tiling $\mathcal{T}_1$ is $\mathcal{T}_{\mathsf{D}(}$$)$. We insert the single chord requirement $G = \{(12, (0, 0))\}$ into $\mathcal{T}_1$, generating the rule

$$\mathcal{T}_1 \xleftarrow{\mathtt{ReqIns}_G} (\mathcal{T}_2, \mathcal{T}_3).$$

The tiling $\mathcal{T}_2$ represents the set of all gridded diagrams with no chords, that is, only the empty gridded diagram of size 0. This is a class whose counting sequence is trivially known; in particular, we apply the verification strategy $V_{\mathcal{T}_2}$ defined previously to get the rule $\mathcal{T}_2 \xleftarrow{V_{\mathcal{T}_2}} ()$. The tiling $\mathcal{T}_3$ represents all the nonempty diagrams of $\mathcal{G}(\mathcal{T}_1)$ and to it we

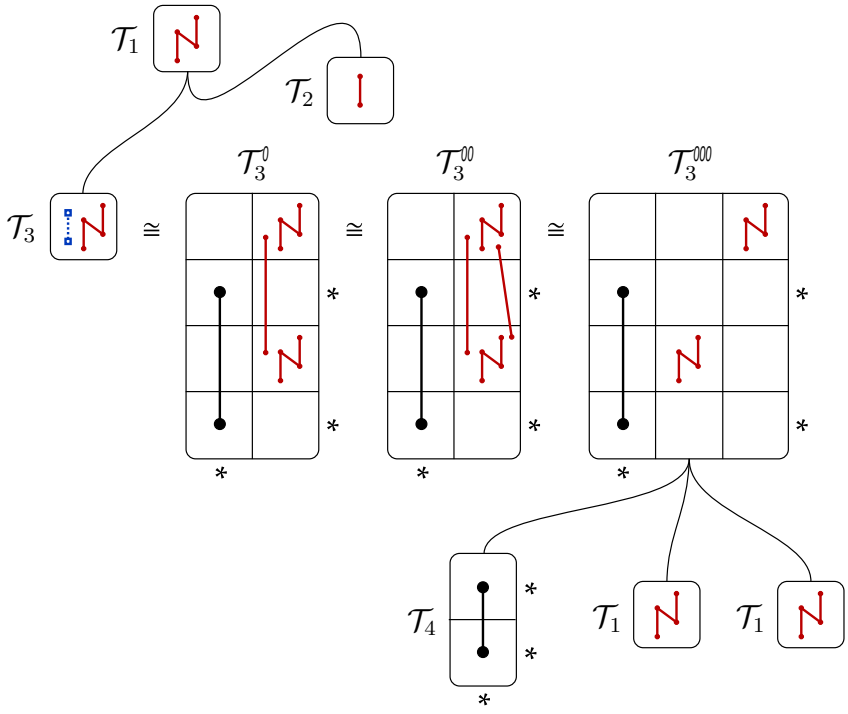


Figure 2.4: A visual representation of the proof tree for noncrossing diagrams $\mathcal{D}(\frown\frown)$.

apply a series of equivalence strategies. Placing the single chord requirement as far left as possible yields the rule

$$\mathcal{T}_3 \xleftarrow{\texttt{ChordPl}_{G,1,2,}} \mathcal{T}_3^{\emptyset}.$$

We then notice that the restricted growth function property and obstruction $(12,((1,1),(1,3)))$ together infer the additional obstruction $H = (21,((1,1),(1,3)))$, generating the rule $\mathcal{T}_3^{\emptyset} \xleftarrow{\texttt{ObsInf}_H} \mathcal{T}_3^{00}$, from which we can then separate column 1 on the nonempty cells, further producing the rule

$$\mathcal{T}_3^{00} \xleftarrow{\texttt{ColSep}_{1,f(1,1)g}} \mathcal{T}_3^{000}.$$

We factor the resulting tiling $\mathcal{T}_3^{000}$ to get the rule

$$\mathcal{T}_3^{000} \xleftarrow{\texttt{Factor}_{Q,(0,0,0),f2,3g}} (\mathcal{T}_4, \mathcal{T}_1, \mathcal{T}_1),$$

where $Q = \{\{(0,0),(0,2)\},\{(1,1)\},\{(2,3)\}\}$, and tilings $\mathcal{T}_4$ and $\mathcal{T}_1$. The former tiling represents only the single chord, which has a trivially verified counting function:

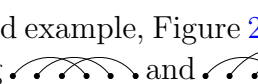
$$\mathcal{T}_4 \xleftarrow{V_{\mathcal{T}_4}} ().$$

Since the latter tiling appears previously in the proof tree this completes the specification. Merging $\mathcal{T}_3$, $\mathcal{T}_3^{\emptyset}$, $\mathcal{T}_3^{\emptyset\emptyset}$, and $\mathcal{T}_3^{\emptyset\emptyset\emptyset}$ into one equivalence class $\mathcal{E}_3$ gives the following specification[4] and corresponding system of equations for the associated generating series.

$$\begin{aligned} \mathcal{T}_1 &\leftarrow (\mathcal{T}_2, \mathcal{E}_3) & T_1(x) &= T_2(x) + E_3(x) \\ \mathcal{T}_2 &\leftarrow () & T_2(x) &= 1 \\ \mathcal{E}_3 &\leftarrow (\mathcal{T}_4, \mathcal{T}_1, \mathcal{T}_1) & E_3(x) &= T_4(x)T_1(x)^2 \\ \mathcal{T}_4 &\leftarrow () & T_4(x) &= x. \end{aligned}$$

This straightforwardly reduces to the equation $T_1 = 1 + xT_1(x)^2$, from which the Catalan generating function follows:

$$T_1(x) = \frac{1 - \sqrt{1-4x}}{2x}.$$

Turning now to our second example, Figure 2.5 shows a proof tree for $\mathcal{C}($  $,$  $)$, connected diagrams avoiding  and  . The root tiling $\mathcal{T}_1 = \mathcal{T}_{\mathcal{C}($  $)}$ must include at least one chord by connectedness, so we infer the single chord requirement and place it as far left as possible to obtain the tiling $\mathcal{T}_2^{\emptyset}$ and equivalence rules

$$\mathcal{T}_1 \xleftarrow{\texttt{ReqListInf}_{fGg}} \mathcal{T}_2 \quad \text{and} \quad \mathcal{T}_2 \xleftarrow{\texttt{ChordPl}_{G,1,2,}} \mathcal{T}_2^{\emptyset}.$$

We then insert a single chord requirement $G^{\emptyset} = \{(12, ((1,1),(1,3)))\}$ into the non-isolated cells, generating the rule

$$\mathcal{T}_2^{\emptyset} \xleftarrow{\texttt{ReqIns}_{G^{\emptyset}}} (\mathcal{T}_3, \mathcal{T}_4).$$

If such a chord does not exist then the linkage across the whole tiling allows us to infer that cells $(1,1)$ and $(1,3)$ are empty, reducing $\mathcal{T}_3$ to the single chord tiling $\mathcal{T}_3^{\emptyset}$ whose counting sequence is known and therefore verifiable as in the previous proof tree. On the other hand, to tiling $\mathcal{T}_4$ we place the required chord leftmost and perform linkage simplification, producing the rule

$$\mathcal{T}_4 \xleftarrow{\texttt{ChordPl}_{G^{\emptyset},1,2,}} \mathcal{T}_4^{\emptyset}.$$

We then factor $T_4^{\emptyset}$, partitioning it into the first chord and the rest of the tiling:

$$\mathcal{T}_{4^{\emptyset}} \xleftarrow{\texttt{Factor}_{Q_1,(0,0),f1g}} (\mathcal{T}_5, \mathcal{T}_3^{\emptyset}),$$

[4]Written without the strategy labels for compactness.

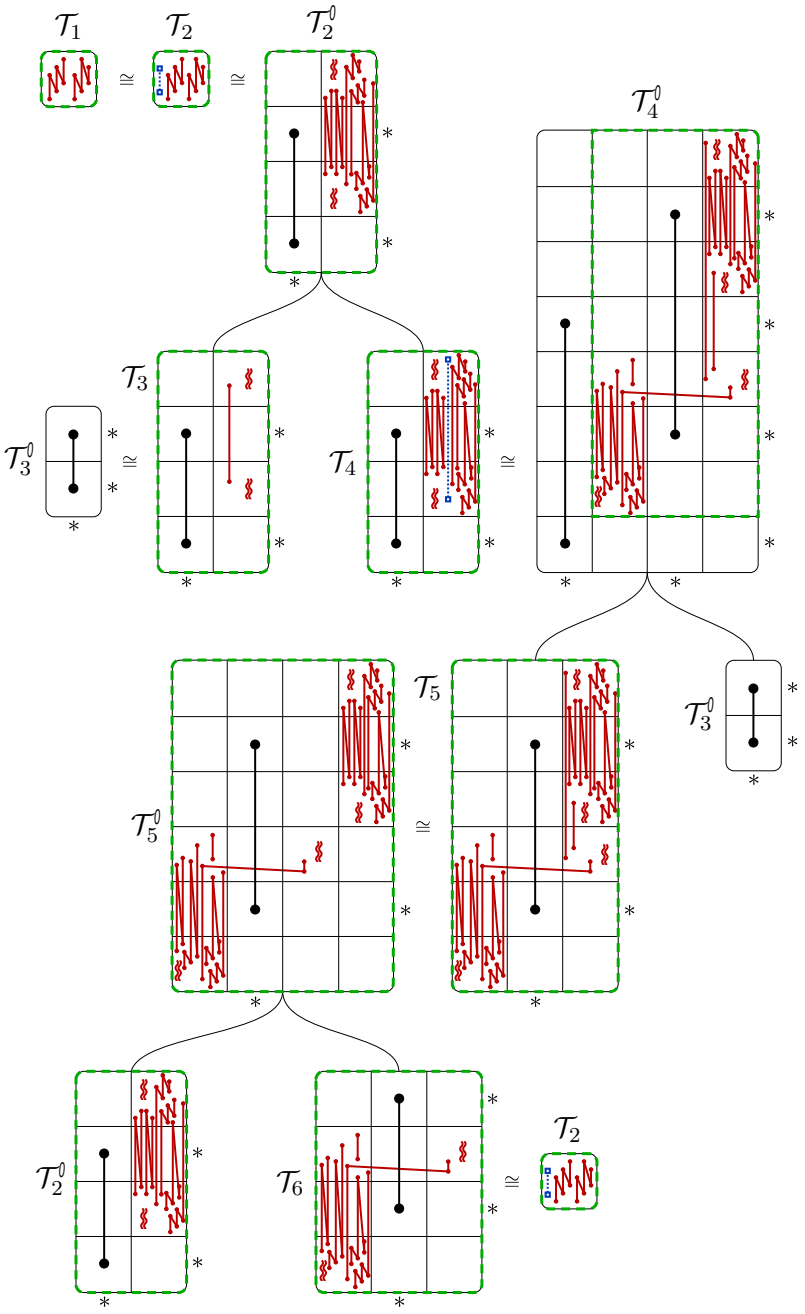


Figure 2.5: A proof tree for the class $\mathcal{C}($  $)$. We save space by using the symbol ≀ to denote the two obstructions in the root tiling $\mathcal{T}_1$.

where $Q_1 = \{\{(1,1),(1,3),(2,2),(2,6),(3,3),(3,5),(3,7)\},\{(0,0),(0,4)\}\}$. While we could have applied the chord placement and factorization steps in reverse order, this would not have allowed for adding and simplifying several obstructions generated during the chord placement—for example, the single chord obstructions between some nonempty cell pairs in the last column are obtained when performing chord placement before factorization, but not vice versa. This illustrates a difficulty with finding proof trees. These single chord obstructions then prove useful, allowing us to perform row separation on the last column after inferring additional obstructions from the restricted growth function property. The resulting tiling $\mathcal{T}_5^{\emptyset}$ can then be factored on the isolated chord in the second column, giving our first application of generalized factorization that actually uses the generalization:

$$\mathcal{T}_5^{\emptyset} \xleftarrow{\texttt{Factor}_{Q_2,(0,1),,;}} (\mathcal{T}_2^{\emptyset}, \mathcal{T}_6).$$

While we could have factored along a partition, not sharing the isolated chord, this would have necessarily generated a tiling that had not appeared before in the proof tree and may not have been prouctively decomposable using the collection of strategies available to us. Instead we get the tiling $\mathcal{T}_2^{\emptyset}$ that has already appeared in the tree and the tiling $\mathcal{T}_6$; observe that the latter can be obtained from $\mathcal{T}_2$ by placing the required chord as far up as possible. Writing $\mathcal{E}_1$, $\mathcal{E}_3$, $\mathcal{E}_4$, and $\mathcal{E}_5$ for the equivalence classes containing tilings $\mathcal{T}_1$, $\mathcal{T}_3$, $\mathcal{T}_4$, and $\mathcal{T}_5$, respectively, we obtain the following specification and system of generating function equations.

$$\begin{aligned}
\mathcal{E}_1 &\leftarrow (\mathcal{E}_3, \mathcal{E}_4) & E_1(x) &= E_3(x) + E_4(x)\\
\mathcal{E}_3 &\leftarrow () & E_3(x) &= x\\
\mathcal{E}_4 &\leftarrow (\mathcal{E}_5, \mathcal{E}_3) & E_4(x) &= E_5(x)E_3(x)\\
\mathcal{E}_5 &\leftarrow (\mathcal{E}_1, \mathcal{E}_1) & E_5(x) &= \frac{1}{x}E_1(x)E_1(x).
\end{aligned}$$

This reduces to $E_1(x) = x + E_1(x)^2$, and solving implies that

$$T_1(x) = E_1(x) = \frac{1-\sqrt{1-4x}}{2} = \frac{C(x)}{x}.$$

We thereby also obtain our first instance of the offset phenomenon for connected classes.

2.3.1 Structural trees

It may have become obvious to the reader at this point that visualizations of proof trees for chord diagram classes can become quite complex and dense, even for the relatively

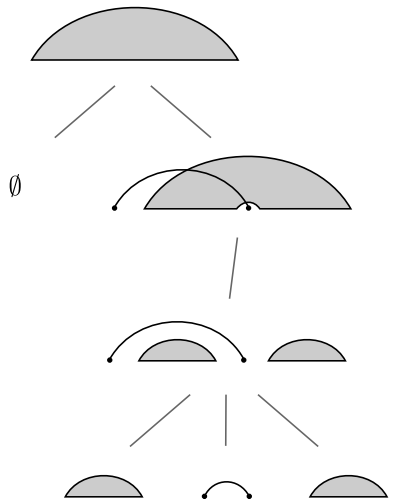


Figure 2.6: A structural tree for the class $\mathcal{D}($$)$.

simple Catalan class $\mathcal{C}($  $)$. While these graphics have the advantage of conveying all the data contained in the tree, often even without accompanying strategy labels, it would be useful to have a heuristic depiction to at-a-glance communicate the most important structural information on a class captured by a proof tree.

We informally encapsulate this idea in the form of what we call a *structural tree*. The vertices of a structural tree roughly correspond to the equivalence classes of a proof tree. For an equivalence class $\mathcal{E}$ of tilings in a given proof tree, a corresponding vertex in an associated structural tree depicts the ungridded structure of a diagram in $\mathcal{G}(\mathcal{T})$ based on the obstructions, requirements, and linkages of some $\mathcal{T} \in \mathcal{E}$. Points and chords are drawn in the usual way for a diagram, while other parts of a diagram are drawn as grey half ovals, emphasizing how their structure is "undiscovered".

Figure 2.6 depicts a structural tree for the class $\mathcal{D}($  $)$ corresponding to the previously given proof tree. Note how in this example in particular the structural tree closely reflects the proof tree itself and matches the standard decomposition of a noncrossing diagram. This is a general feature—structural trees essentially emulate how a human might discover a proof tree when decomposing a diagram class by hand.

Figure 2.7 displays a structural tree for the class $\mathcal{C}($  $)$ which also corresponds to the previously given proof tree for this class. Here the utility of structural trees starts to become apparent. While the proof tree significantly grows in complexity from the previous class to this one, with many more obstructions that are difficult to distinguish when visualized, the structural tree does not, mainly only increasing in size as

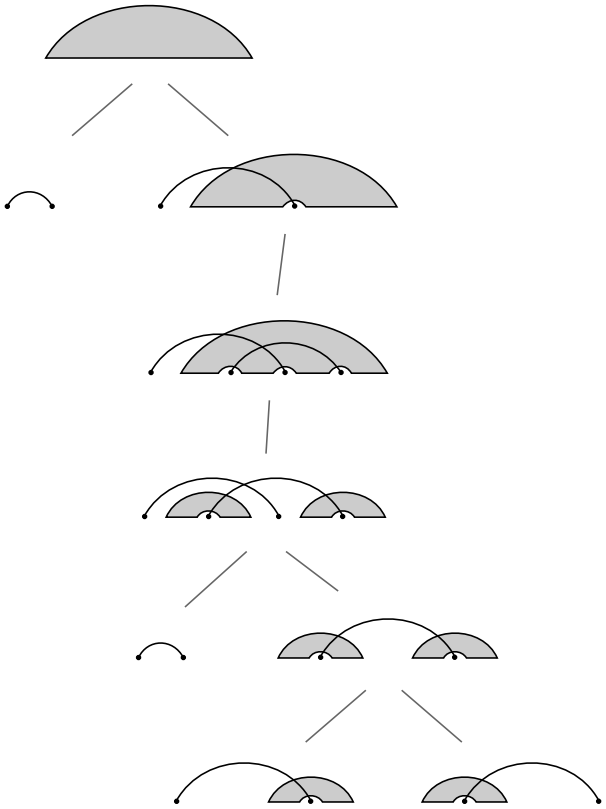


Figure 2.7: A structural tree for the class $\mathcal{C}($  $)$.

a tree.

We will use structural trees to both aid understanding of a proof tree and, when the proof tree is too large to be displayed in an effective manner, provide a proxy for it in a proof.

Chapter 3

Enumerating chord diagram classes

In this chapter we enumerate a variety of classes of the form $\mathcal{D}(\mathcal{X})$, $\mathcal{C}(\mathcal{X})$, $\mathcal{T}(\mathcal{X})$, and $\mathcal{ST}(\mathcal{X})$, each defined by forbidding a set of diagrams $\mathcal{X}$ as subdiagrams and possibly imposing one of the three connectedness notions: connectivity, 1-terminality, and 1-sym-terminality. Roughly speaking, we begin by considering classes amenable to the structural decomposition framework developed in Chapter 2. We then transition to studying classes that either require other enumeration techniques or for which explicit bijections can be obtained in addition to a closed-form expression for the counting sequence or generating series.

The class $\mathcal{C}(T_{>3}, B_{>3})$ of tree diagrams was first counted by Leroux and Miloudi [71] and, shortly thereafter, by Dulucq and Penaud [38]. We begin by reproving their result using Combinatorial Exploration, primarily as an initial demonstration of the framework applied to a class avoiding an infinite set of diagrams. Unlike their approaches, our method has the advantage of working directly on the diagrams, in particular avoiding passing to other objects or employing sophisticated algebraic enumeration techniques.[1] Stoimenow [99] also obtained this result directly using a closely related chord diagram decomposition.

Proposition 3.0.1 (Leroux and Miloudi [71], Dulucq and Penaud [38])**.** *The generating series and counting sequence of* $\mathcal{C}(T_{>3}, B_{>3})$ *are*

$$\frac{A(x)}{\sqrt[3]{18}} + \frac{\sqrt[3]{2/3}x}{A(x)} \quad \textit{and} \quad \frac{1}{2n-1}\binom{3n-3}{n-1}, \tag{3.1}$$

where $A(x) = \sqrt[3]{\sqrt{3}\sqrt{27x^4 - 4x^3} - 9x^2}$.

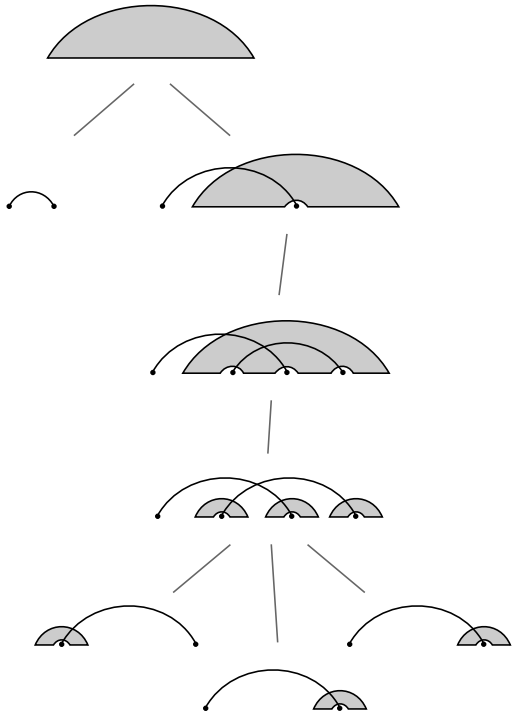


Figure 3.1: A structural tree for the class $\mathcal{C}(T_{>3}, B_{>3})$.

Proof. In Figures 3.2 and 3.1 we give a proof tree and corresponding structural tree for $\mathcal{C}_{n+1}(T_{>3}, B_{>3})$. Writing $\mathcal{E}_1$, $\mathcal{E}_3$, an $\mathcal{E}_4$ for the equivalence classes containing tilings $\mathcal{T}_1$, $\mathcal{T}_3$, and $\mathcal{T}_4$, we have the following combinatorial specification and corresponding system of generating function equations:

$$\begin{aligned}
\mathcal{E}_1 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_3, \mathcal{E}_4) && E_1(x) = E_3(x) + E_4(x) \\
\mathcal{E}_3 &\xleftarrow{V_{E_3}} () && E_3(x) = x \\
\mathcal{E}_4 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_1, \mathcal{E}_1, \mathcal{E}_1) && E_4(x) = \frac{1}{x} E_1(x)^3
\end{aligned}$$

We indicate the combinatorial strategies corresponding to each rule but omit the parameters, which are derivable from the proof tree. We briefly note though that transforming tiling $\mathcal{T}_4^{\emptyset}$ into the usefully factorable tiling $\mathcal{T}_4^{000}$ involves our first two applications of the cell

[1] In particular, Pólya theory in the case of Leroux and Miloudi [71].

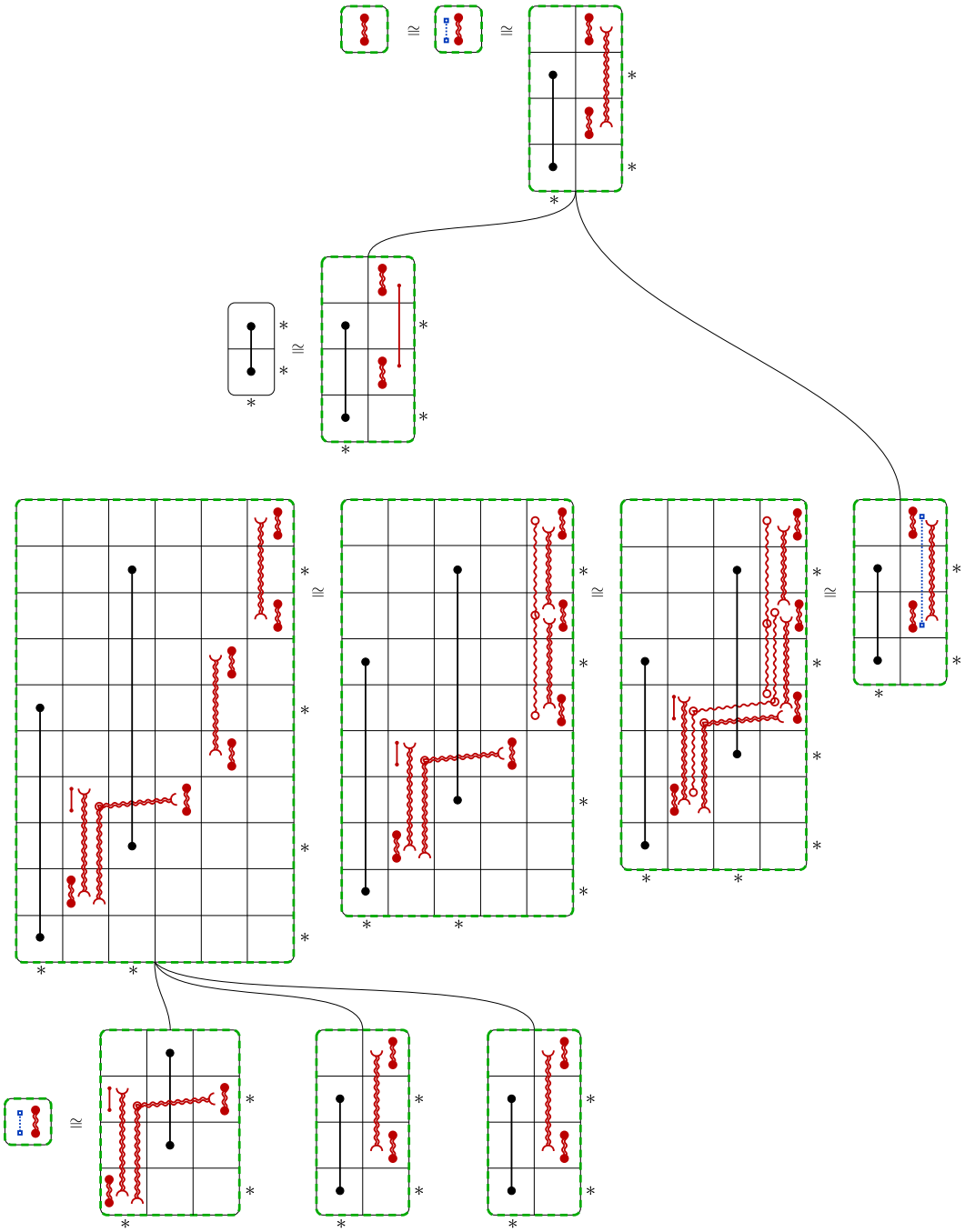


Figure 3.2: A proof tree for the class $\mathcal{C}(T_{>3}, B_{>3})$ of tree diagrams.

splitting strategy. From the system of equations it follows that

$$T_1(x) = x + \frac{1}{x} T_1(x)^3.$$

It is well known and verifiable using any computer algebra system that solving this gives the desired generating series and counting sequence. □

Dulucq and Penaud [38] first proved this result by exhibiting a bijection between tree diagrams and ternary trees, which can easily be obtained recursively from our decomposition or by directly mapping each crossing to an internal vertex and (all but the initial) point to a leaf. Many other combinatorial objects also have the same counting sequence (see [93, A001764]). Kreweras [69] first defined the well-known lattice $\mathcal{L}_n^K$ of noncrossing partitions of size n ordered by refinement and showed that the cardinality of its set $\mathrm{Int}(\mathcal{L}_n^K)$ of intervals is given by (3.1), with n replaced by $n+1$. In 2020, Colin Defant [35] proved that Kreweras intervals are in bijection with certain pattern-avoiding permutations constructed from West's stack-sorting map s, a variant of the stack-sorting algorithm introduced by Knuth [67] and studied extensively in West's Ph.D. thesis [107]. There has been considerable interest in this map, especially with regards to permutations σ with positive *fertility*, that is, $|s^{-1}(\sigma)|$, the number of preimages of σ under s. Such permutations are referred to as *sorted*, and they are *uniquely sorted* if the fertility is 1. Let $\mathcal{U}_n$ denote the set of uniquely sorted permutation in $\mathcal{S}_n$; it was proved in [36] that U_n is empty if n is even. Defant proved that $\mathrm{Int}(\mathcal{L}_n^K)$ is in bijection with permutations of $\mathcal{U}_{2n+1}$ that avoid the patterns 312 and 1342.

We summarize these relations in the following corollary. Together with the next result, this shows that the offset phenomenon applies to the class of tree diagrams.

Corollary 3.0.2 (Dulucq and Penaud [38], Kreweras [69], Defant [36])**.** *Tree diagrams of size $n+1$ are equinumerous with*

- *complete ternary trees with n internal vertices,*
- *intervals of the n^{th} Kreweras lattice of size n, and*
- *$(312, 1342)$-avoiding uniquely sorted permutations of length $2n+1$.*

It remains an open problem to find bijections between tree diagrams and the latter two combinatorial objects. There is at least one other diagram class which is equinumerous with $\mathcal{C}(T_{>3})$ and enumerable using our strategy pool.

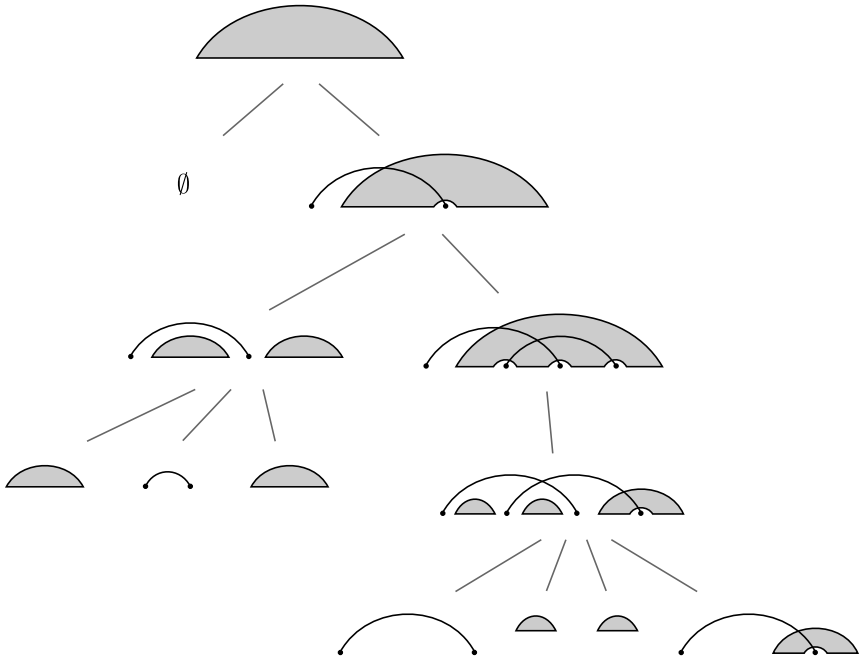


Figure 3.3: A structural tree for the class $\mathcal{D}($  $)$.

Theorem 3.0.3. *The generating series and counting sequence of*

$\mathcal{D}($  $)$

are

$$\frac{A(x)}{\sqrt[3]{18}x} + \frac{\sqrt[3]{2/3}}{A(x)} \quad \textit{and} \quad \frac{1}{2n+1}\binom{3n}{n},$$

where $A(x)$ is as de ned previously.

Proof. We display in Figure 3.3 a structural tree for this class. For brevity, we omit the corresponding proof tree composed of tiling equivalence classes $\mathcal{E}_1$, $\mathcal{E}_2$, $\mathcal{E}_3$, $\mathcal{E}_4$, $\mathcal{E}_5$, and $\mathcal{E}_6$; the correspondence between the two trees is closely analogous to the correspondence between previous pairs of proof trees and their structural proxies. Nevertheless, the combinatorial specification and corresponding set of generating function equations are as follows:

$$\mathcal{E}_1 \xleftarrow{\texttt{ReqIns}} (\mathcal{E}_2, \mathcal{E}_3) \qquad E_1(x) = E_2(x) + E_3(x)$$

$$\mathcal{E}_2 \xleftarrow{V_{E_2}} () \qquad E_2(x) = 1$$

$$\begin{aligned}
\mathcal{E}_3 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_4, \mathcal{E}_6) && E_3(x) = E_4(x) + E_6(x)\\
\mathcal{E}_4 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_1, \mathcal{E}_5, \mathcal{E}_1) && E_4(x) = E_5(x)E_1(x)^2\\
\mathcal{E}_5 &\xleftarrow{V_{E_5}} () && E_5(x) = x\\
\mathcal{E}_6 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_5, \mathcal{E}_1, \mathcal{E}_1, \mathcal{E}_3) && E_6(x) = E_5(x)E_1(x)^2E_3(x)
\end{aligned}$$

This reduces to the system

$$\begin{aligned}
E_1(x) &= 1 + E_3(x)\\
E_3(x) &= xE_1(x)^2 + xE_1(x)^2E_3(x),
\end{aligned}$$

which can be solved with a computer algebra system to infer the result. □

Corollary 3.0.4. *The sets $\mathcal{C}_{n+1}(T_{>3})$ and $\mathcal{D}_n($*  *$)$ are equinumerous.*

3.1 Avoiding bottom cycles

This section is based on unpublished joint work with Ali Assem Mahmoud. In this section, we explicitly enumerate the class $\mathcal{C}(B_{>3})$ of connected bottom-cycle-free diagrams using the Combinatorial Exploration framework.

Theorem 3.1.1. *The generating function of $\mathcal{C}(B_{>3})$ is*

$$\frac{3x - x\sqrt{1-8x}}{2(x+1)},$$

implying that

$$|\mathcal{C}_{n+1}(B_{>3})| = \frac{1}{n}\sum_{k=0}^{n-1}\binom{2n}{n-1-k}\binom{n-1+k}{k},$$

the n^{th} generalized Catalan number $\hat{C}_n$.

Proof. Figure 3.4 gives a structural tree for this class. We also include the corresponding proof tree in Figure A.1 of Appendix A; it is too large to reasonably display within this section. Writing $\mathcal{E}_1$, $\mathcal{E}_3$, $\mathcal{E}_4$, $\mathcal{E}_5$, $\mathcal{E}_7$, $\mathcal{E}_8$, and $\mathcal{E}_{10}$ for the equivalence classes of the tilings in

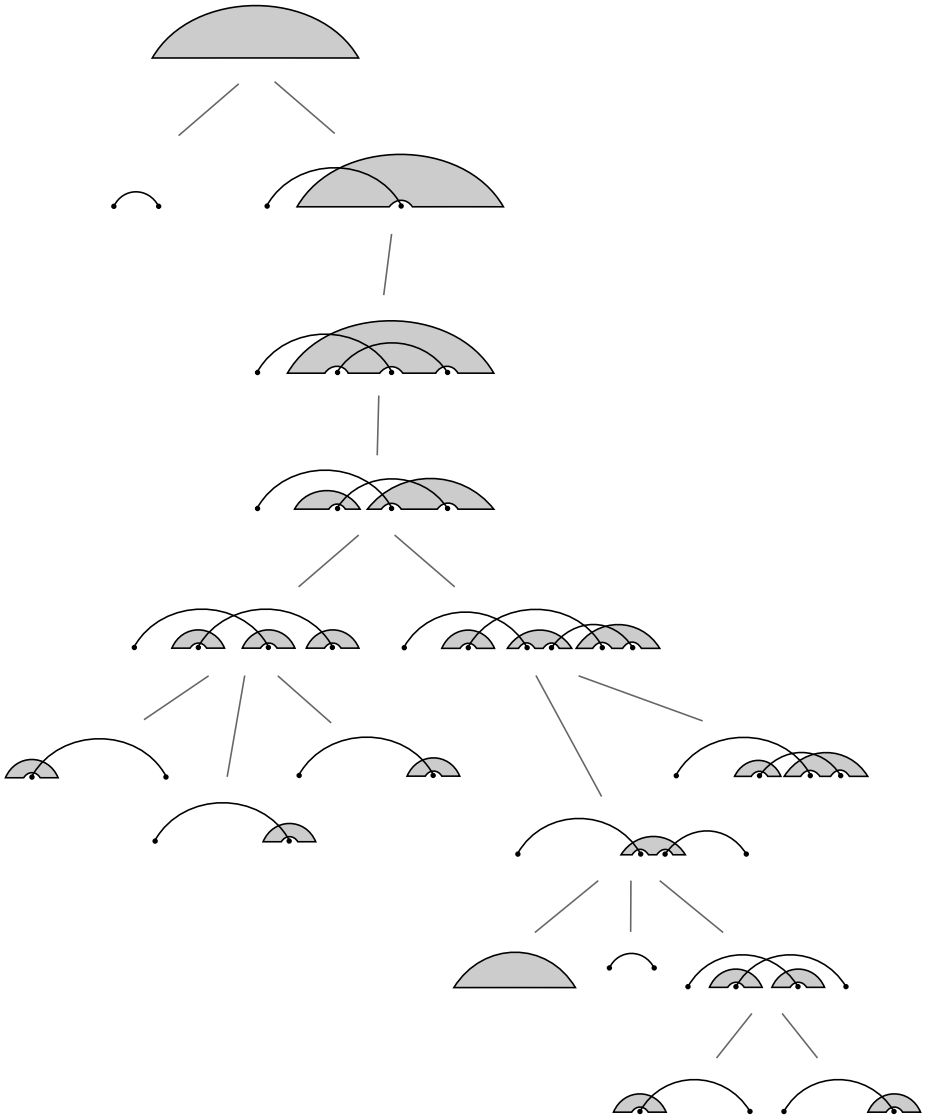


Figure 3.4: A structural tree for the class $\mathcal{C}(B_{>3})$ of connected bottom-cycle-free diagrams.

the proof tree, the combinatorial specification and corresponding set of generating function equations are as follows:

$$\mathcal{E}_1 \xleftarrow{\texttt{ReqIns}} (\mathcal{E}_3, \mathcal{E}_4) \qquad E_1(x) = E_3(x) + E_4(x)$$

$$\mathcal{E}_3 \xleftarrow{V_{E_3}} () \qquad E_3(x) = x$$

$$\begin{aligned}
\mathcal{E}_4 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_5, \mathcal{E}_7) & E_4(x) &= E_5(x) + E_7(x)\\
\mathcal{E}_5 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_1, \mathcal{E}_1, \mathcal{E}_1) & E_5(x) &= \frac{1}{x}E_1(x)^3\\
\mathcal{E}_7 &\xleftarrow{\texttt{Factor}} (E_8, E_4) & E_7(x) &= \frac{1}{x}E_8(x)E_4(x)\\
\mathcal{E}_8 &\xleftarrow{\texttt{Subtract}} (E_1, E_3, E_{10}) & E_8(x) &= E_1(x) - E_3(x) - E_{10}(x)\\
\mathcal{E}_{10} &\xleftarrow{\texttt{Factor}} (E_1, E_1) & E_{10}(x) &= E_1(x)^2
\end{aligned}$$

Here we see our first application of the cell shuffling strategy enabling converting tiling $\mathcal{T}^9$ to the equivalent $\mathcal{T}_4^{\emptyset\emptyset}$ (in the structural tree, compare the the seventh internal vertex to its second child); structurally, it is used to detach the subdiagram attached to the source of the root and reattach it to the source of its first neighbor.

The proof tree presented here uses a new subtraction strategy, `Subtract`, whose general definition we leave to later work. This strategy relates to the extensive literature on defining a combinatorial operation of subtraction, especially in the context of species. A full solution of this problem was obtained by Joyal [61, 60] and Yeh [112, 113] using virtual species. In this case, the strategy expresses tiling $\mathcal{T}_8$ as the subtraction of tilings $\mathcal{T}_3^{\emptyset}$ and $\mathcal{T}_{10}$ from the supertiling $\mathcal{T}_2^{\emptyset}$. That is, after applying certain transforming equivalence strategies, every gridded diagram in $\mathcal{G}(\mathcal{T}_2^{\emptyset})$ can be realized as a gridded diagram in either $\mathcal{G}(\mathcal{T}_8)$, $\mathcal{G}(\mathcal{T}_3^{\emptyset})$, or $\mathcal{G}(\mathcal{T}_{10})$; in particular, modulo tiling equivalence,

$$\mathcal{G}(\mathcal{T}_8) = \mathcal{G}(\mathcal{T}_2^{\emptyset}) - (\mathcal{G}(\mathcal{T}_3^{\emptyset}) \sqcup \mathcal{G}(\mathcal{T}_{10})),$$

where $\sqcup$ denotes disjoint union. While this strategy does not satisfy the definition of productivity because $|\mathcal{G}_n(\mathcal{T}_2^{\emptyset})| > |\mathcal{G}_n(\mathcal{T}_8)|$ for all n, it nevertheless produces a productive proof tree in this case.[2] This follows from the fact that the above system of generating function equations has a unique solution, as we now observe. In particular, the system straightforwardly reduces by substitution to the pair of equations

$$\begin{aligned}
E_1(x) &= x + E_4(x)\\
E_4(x) &= \frac{1}{x}E_1(x)^3 + \frac{1}{x}E_4(x)(E_1(x) - x - E_1(x)^2),
\end{aligned}$$

and a computer algebra system solves this for the desired generating function. □

[2]For species this case corresponds to subtracting subspecies, for which it is much easier to show that it is a suitably well-behaved operation on species compatible with generating functions and combinatorial specifications.

See [93, A064062] for further information on the generalized Catalan numbers. Note that these numbers also count $(2413, 3142)$-avoiding Dumont permutations [18]; see [39] for a definition of these kind of permutations. Bloom and Elizalde [8] proved that this sequence also counts most of the classes obtained by excluding a pair of permutation diagrams of size three. Bloom and Elizalde obtained this result by passing to rook placements on Ferrers boards and counting these objects by size and another parameter. Here, we reprove their result for the class

$\mathcal{D}($  $)$

directly on the level of diagrams[3] using a compact decomposition obtained in the Combinatorial Exploration framework. Unlike for the class $\mathcal{C}(B_{>3})$ it does not require the cell splitting or cell shuffling strategies, nor a subtraction strategy.

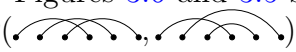
Theorem 3.1.2 (Bloom and Elizalde [8])**.** *Fix*

$$(\sigma, \tau) \in \{(123, 213), (123, 132), (321, 312), (231, 312), \\ (231, 213), (231, 132), (312, 213), (312, 132), (213, 132)\}.$$

Then the generating function of $\mathcal{D}(D_\sigma, D_\tau)$ *is*

$$\frac{3 - \sqrt{1 - 8x}}{2(x + 1)},$$

implying that $|\mathcal{D}_n(D_\sigma, D_\tau)| = \hat{C}_n$.

Proof for $\sigma = 123$ *and* $\tau = 132$. Figures 3.6 and 3.5 show a proof tree and corresponding structural tree for the class $\mathcal{D}($  $)$. The corresponding combinatorial specification and system of generating function equations is as follows:

$$\begin{aligned}
\mathcal{E}_1 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_2, \mathcal{E}_3) && E_1(x) = E_2(x) + E_3(x) \\
\mathcal{E}_2 &\xleftarrow{V_{E_2}} () && E_2(x) = 1 \\
\mathcal{E}_3 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_4, \mathcal{E}_6) && E_3(x) = E_4(x) + E_6(x) \\
\mathcal{E}_4 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_5, \mathcal{E}_1, \mathcal{E}_1) && E_4(x) = E_5(x)E_1(x)^2 \\
\mathcal{E}_5 &\xleftarrow{V_{E_5}} () && E_5(x) = x
\end{aligned}$$

[3] Or, really, gridded diagrams.

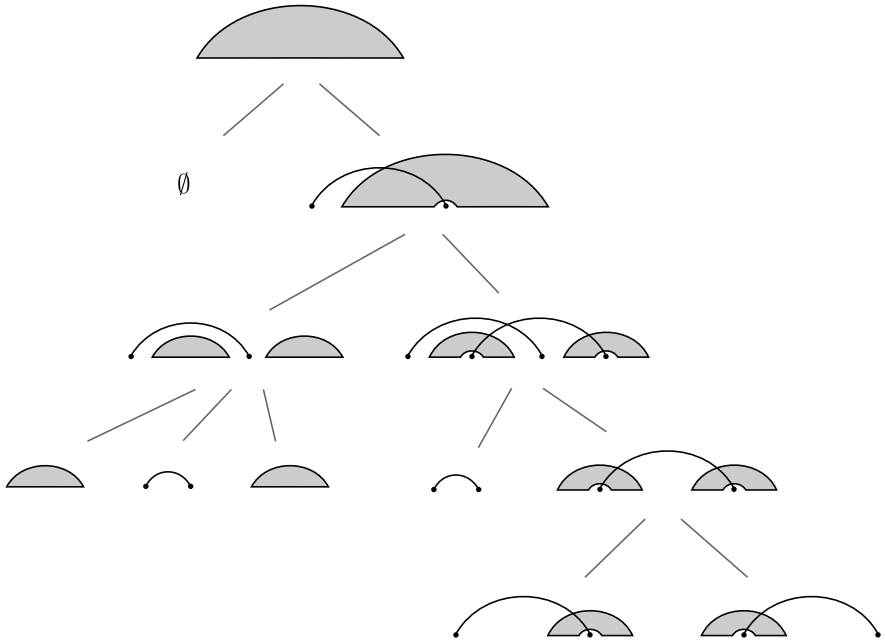


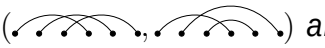
Figure 3.5: A structural tree for the class $\mathcal{D}($  $)$.

$$\mathcal{E}_6 \xleftarrow{\texttt{Factor}} (E_7, E_5) \qquad E_6(x) = E_7(x)E_5(x)$$
$$\mathcal{E}_7 \xleftarrow{\texttt{Factor}} (E_3, E_3) \qquad E_7(x) = \frac{1}{x}E_3(x)^2$$

This reduces to the pair of equations

$$E_1(x) = 1 + E_3(x)$$
$$E_3(x) = xE_1(x)^2 + E_3(x)^2,$$

and applying a computer algebra system gives the desired generating function solution. □

Corollary 3.1.3. *The sets $\mathcal{C}_{n+1}(B_{>3})$ and $\mathcal{D}_n($*  *$)$ are equinumerous.*

As is universally the case thus far, this implies that $\mathcal{C}(B_{>3})$ is another connected class exhibiting the offset phenomenon. We have thus far been unable to obtain a bijection explaining the enumerative equivalence between these two classes, although we will comment further on this possibility in later sections.

We now preview applying Combinatorial Exploration to 1-terminal diagram classes by using it to enumerate the class $\mathcal{T}(B_{>3})$ of 1-terminal bottom-cycle-free diagrams. In later

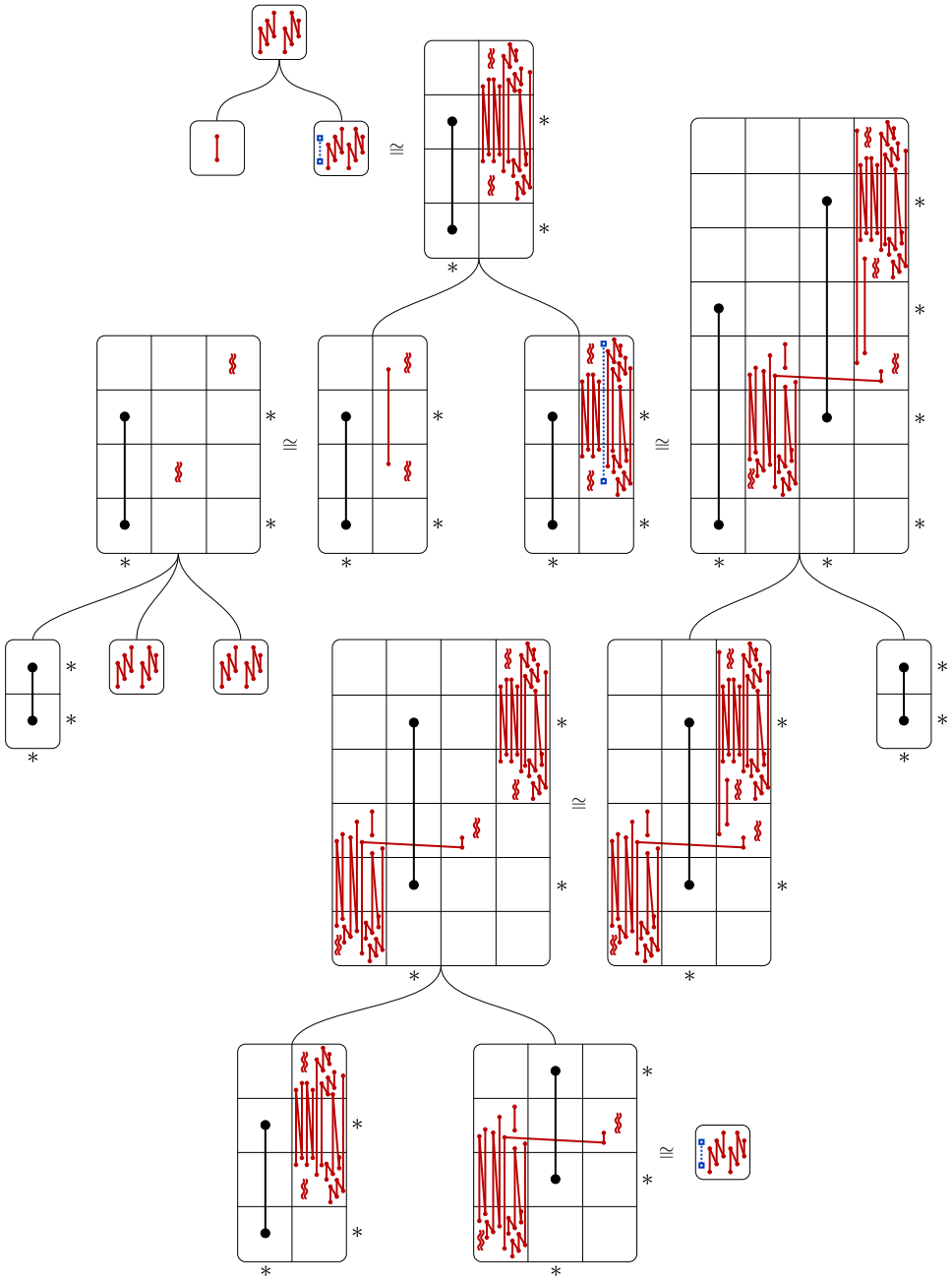


Figure 3.6: A proof tree for the class $\mathcal{D}($  $)$. The symbol ≀ denotes the two obstructions in the root tiling $\mathcal{T}_1$.

work we will formally extend Combinatorial Exploration to 1-terminal classes. The only significant difference is that 1-terminal linkages can be used to infer more structure than connected linkages.

Proposition 3.1.4. *The generating function of $\mathcal{T}(B_{>3})$ is*

$$\frac{3x - x^2 - x\sqrt{1 - 6x + x^2}}{2},$$

implying that

$$|\mathcal{T}_{n+2}(B_{>3})| = \sum_{k=0}^{n} C_k \binom{n+k}{n-k},$$

the n^{th} large Schroder number S_n.

Proof. We give a structural tree for this class in Figure 3.7. The underlying proof tree proceeds similarly to the connected counterpart while taking advantage of additional obstructions and simplifications enabled by 1-terminality. The resulting combinatorial specification and system of equations are given below:

$$\begin{aligned}
\mathcal{E}_1 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_3, \mathcal{E}_4) & E_1(x) &= E_3(x) + E_4(x) \\
\mathcal{E}_3 &\xleftarrow{V_{E_3}} () & E_3(x) &= x \\
\mathcal{E}_4 &\xleftarrow{\texttt{ReqIns}} (\mathcal{E}_5, \mathcal{E}_7) & E_4(x) &= E_5(x) + E_7(x) \\
\mathcal{E}_5 &\xleftarrow{\texttt{Factor}} (\mathcal{E}_1, \mathcal{E}_3, \mathcal{E}_1) & E_5(x) &= E_1(x)^2 \\
\mathcal{E}_7 &\xleftarrow{\texttt{Factor}} (E_8, E_4) & E_7(x) &= \frac{1}{x} E_8(x) E_4(x) \\
\mathcal{E}_8 &\xleftarrow{\texttt{Subtract}} (E_1, E_3, E_{10}) & E_8(x) &= E_1(x) - E_3(x) - E_{10}(x) \\
\mathcal{E}_{10} &\xleftarrow{\texttt{Factor}} (E_1, E_3) & E_{10}(x) &= E_1(x) E_3(x)
\end{aligned}$$

This reduces to the pair of equations

$$\begin{aligned}
E_1(x) &= x + E_4(x) \\
E_4(x) &= E_1(x)^2 + \frac{1}{x} E_4(x)(E_1(x) - x - xE_1(x)),
\end{aligned}$$

which can be solved with a computer algebra system to give the desired generating function. □

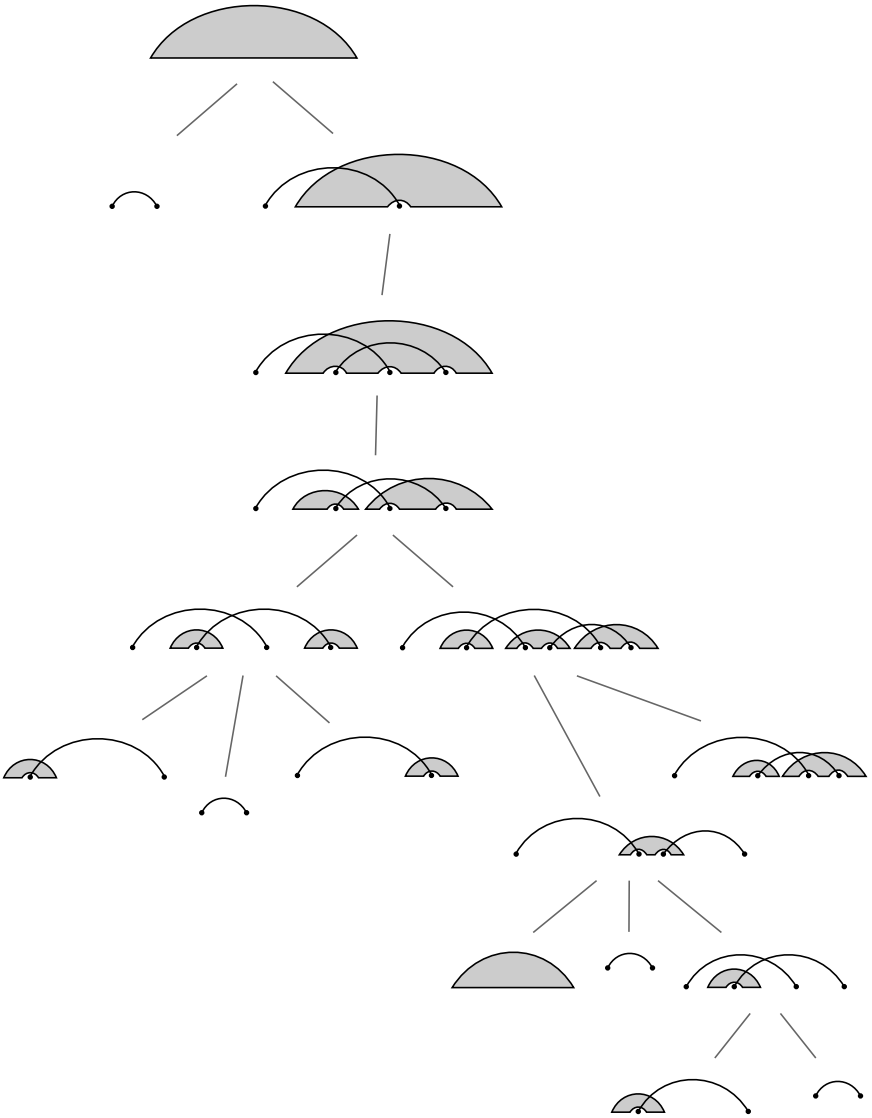


Figure 3.7: A structural tree for the class $\mathcal{T}(B_{>3})$ of 1-terminal bottom-cycle-free diagrams.

The large Schröder numbers obey the Catalan-like recurrence

$$S_n = S_{n\ \ 1} + \sum_{k=1}^{n\ \ 1} S_k S_{n\ \ k}$$

and are the counting sequence of a large variety of combinatorial objects (see [93, A006318]),

as are the little Schröder numbers $s_n = S_n/2$ [93, A001003]. The following corollary records these connections and shows that the offset phenomenon applies to $\mathcal{T}(B_{>3})$. See [93, A006318] or the cited references for definitions.

Corollary 3.1.5. *1-terminal bottom-cycle-free diagrams of size $n+2$ are equinumerous with*

- *Schröder paths of semilength n,*
- *$(2413, 3142)$-avoiding permutations of length $n+1$, and*
- *semi-standard Young tableaux of shape $n \times 2$ with consecutive entries.*

3.2 Avoiding top cycles

The work in the following two sections, as well as the first two sections of Chapter 4, is based on the paper [79] of the author.

We now turn to enumerating connected diagrams avoiding top cycles, $\mathcal{C}(T_{>3})$, as well as related classes. While the decomposition strategies developed in Section 2 have proven successful when applied to classes excluding bottom cycles, they do not appear to be effective for top-cycle-free classes. We have thus far not been able to find a proof tree for $\mathcal{C}(T_{>3})$ with our strategy collection. It remains possible though that one exists, in particular because we have yet to conduct a suitable computer search. Furthermore, even failing that, it is likely that further strategies could be described and added to our collection that would yield a solution to this enumeration problem. We leave exploring this possibility to later work.

Instead, we develop a novel method for decomposing connected diagrams. While as formalized here it is entirely separate from the Combinatorial Exploration framework, it is based on structural ideas related to several combinatorial strategies. The method involves essentially partitioning a connected chord diagram into specific subdiagrams. This partition is constructed by first partitioning the 1-terminal part of the diagram revealed by Lemma 1.3.5 into certain 1-terminal subdiagrams and then bootstrapping the rest of the construction onto these 1-terminal pieces. The resulting decomposition applies to any connected diagram but is particularly well-behaved for top-cycle-free diagrams.

For the rest of the section, let C be a connected diagram of size n.

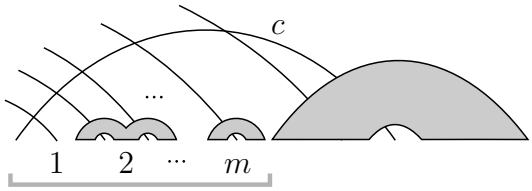


Figure 3.8: A representative visualization of the source-sink group, indicated by the grey bracket, of a chord c in a connected diagram of valency m.

3.2.1 Grouping and tracing

Write $N_L(c)$ and $N_R(c)$ for the left and right neighbors, respectively, of a chord $c \in C$.

Definition 3.2.1 (Source-sink group)**.** We say that a chord $c \in C$ is *attached* to an indecomposable subdiagram B of C if $\{c\}$ is not a connected component of $B \cup \{c\}$; symmetrically, B is also *attached* to c. Define the *source-sink group* of c to be the interval of endpoints containing:

- the source of c,
- the indecomposable components of $C - N_L(c)$ nested under c and not nested under any chord $c^{\emptyset} > c$, and
- the sinks of $N_L(c)$ whose associated chord is either attached to one of the above indecomposable components or does not cross any chord $c^{\emptyset} > c$.

The source-sink groups of C refers to the source-sink groups of the chords of C.

See Figure 3.8. More informally, we will also refer to chords as "attached" to their sources and sinks. Observe that that the source-sink group of c can also be characterized as the points between the source of c (inclusive) and the source of the next chord (exclusive) in the intersection order; while this characterization is simpler, it does not offer the local structural information of the above definition. Note that there may be no sinks in a source-sink group. Source-sink groups enable us to define a notion of *valency* for chords in a diagram.

Definition 3.2.2. Define the *valency* of $c \in C$ to be $k + \ell$ if there are exactly k left neighbors of c not attached to any chord $c^{\emptyset} > c$ and after removing the left neighbors of c we are left with exactly ℓ indecomposable components immediately following the source of c.

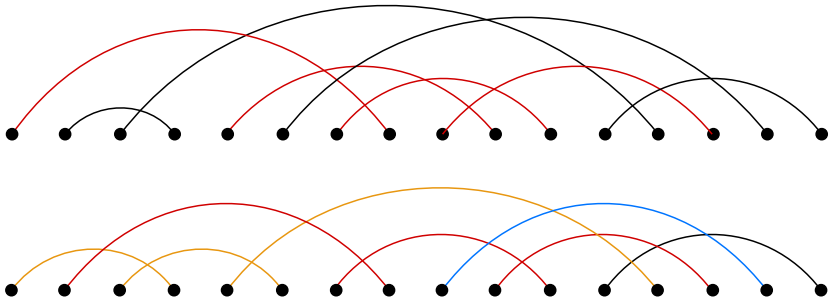


Figure 3.9: Above: a 1-terminal diagram with the traced subdiagram of the rightmost red chord indicated in red. Below: another 1-terminal diagram with the traced subdiagrams of the neighbors of the terminal chord colored.

Loosely speaking, in a local sense the valency is the number of connected pieces in the source-sink group of a chord. This notion of valency for a diagram should be thought of as playing the same role as outdegree for a tree, that is, the number of children of a vertex in a rooted tree. The chord in Figure 3.8 has valency m. As we will see, our decomposition preserves source-sink groups and therefore valencies. This will in particular gain importance in Chapter 4.

Let T be a 1-terminal diagram. By Lemma 1.3.4 the source-sink groups of T contain only a source and adjacent maximal interval of sinks. The source-sink group of a chord $c \in T$ contains no sinks precisely when c is the only chord of T or there is another source following the source of c.

Definition 3.2.3 (Traced subdiagram)**.** For $c \in T$, we construct the *traced subdiagram* T_c *of* c as follows. The chord c belongs to T_c. Beginning with $c^{\emptyset} = c$, place in T_c the chord attached to every sink in the source-sink group of $c^{\emptyset}$, and then repeat this procedure with each newly added chord of T_c.

See Figure 3.9 for examples. Since for 1-terminal diagrams every chord attached to a sink in the source-sink group crosses the chord of the source to the left, traced subdiagrams are equivalently specified as follows: each chord of D besides the base chord c is contained in T_c if and only if its last right neighbor (in the standard order) is in T_c. In particular, we observe the following.

Lemma 3.2.4. *Traced subdiagrams T_c are 1-terminal and c is the terminal chord.*

The above observation together with the following result gives a unique decomposition

of a 1-terminal diagram into smaller 1-terminal subdiagrams, after removing the terminal chord.

Lemma 3.2.5. *The traced subdiagrams of the neighbors of the terminal chord d of T partition $T - d$.*

Proof. If $|T| = 1$, the result trivially holds. Otherwise, the root chord c is either a neighbor of d or it is not. If it is, note that by 1-terminality there are no chords nested under the terminal chord. Then clearly $T_c = \{c\}$ and, since the last right neighbor of c is d, it is not contained in any other traced subdiagram besides T_d. Since traced subdiagrams are 1-terminal, by Corollary 1.3.6 it follows that all other traced subdiagrams (excluding T_d) are inherited from $T - c$. Then we get the required partition inductively or immediately if $T - c - d$ is empty. On the other hand, if c is not a neighbor of d then it is contained in the traced subdiagram of T of its rightmost neighbor, which exists since T is connected. Then, as before, inductively we get the required partition. □

Although we will not need this fact, it is also worth observing that this implies that the traced subdiagram of the terminal chord of T is the entire diagram.

3.2.2 A recursive bijection

We are now ready to construct our decomposition of connected diagrams. The construction comes in the form of a map α sending each connected diagram C to a tuple of connected diagrams along with a partition of an interval $[j]$. The partition specifies how the connected diagrams are shuffled together to generate C.

Let $c_1 \prec c_2 \prec \cdots \prec c_n$ be the chords of C in the intersection order and $j + 1$ be the index $t_1(C)$ of the first terminal chord of C.[4]

Definition 3.2.6. We define the following.

- T: the 1-terminal diagram $\{c_1, \ldots, c_{j+1}\}$.
- $A_1, A_2, \ldots, A_k$: the indecomposable components of $C - \{c_1, \ldots, c_j\}$ nested under c_{j+1}. Neighbors of the terminal chord c_{j+1} may or may not be attached to such a component; for each that is not, we also include among these A_ℓ's an empty component and consider them attached to their paired neighbor chord.

[4] That is, the chord with rightmost sink.

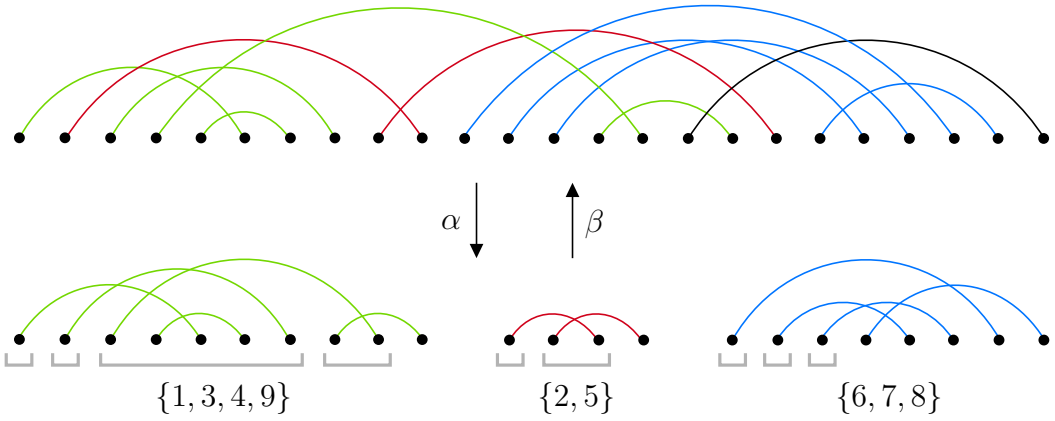


Figure 3.10: A connected diagram and its permuted decomposition defining the maps α and β. The source-sink groups used in the construction of β are indicated by the brackets below the three diagrams on the bottom of the figure.

- For all $1 \leq \ell \leq k$, D_ℓ: the union of the traced subdiagrams of T of each neighbor of c_{j+1} attached to component A_ℓ in C; if A_ℓ is empty this is simply the traced subdiagram of the neighbor defining A_ℓ above.
- For all $1 \leq \ell \leq k$, C_ℓ: the union of D_ℓ with all indecomposable components of $C - T$ attached to chords of D_ℓ in C.
- For all $1 \leq \ell \leq k$, I_ℓ: the subset $\{1 \leq i \leq j \mid c_i \in C_\ell\}$ of $[1, j]$ indicating which chords in $T - c_{j+1}$ belong to C_ℓ. Furthermore, set $i_\ell = |I_\ell|$.
- $\alpha(C) = ((C_1, I_1), (C_2, I_2), \ldots, (C_k, I_k))$.

Figure 3.10 illustrates an example of this construction for a representative diagram C. By Lemma 1.3.4 chords of T attach to indecomposable components of $C - T$ at sinks; in other words, the interval occupied by an indecomposable component of $C - T$ intersects at most one maximal interval of sinks of T. By the construction of traced subdiagrams, it follows that each indecomposable component of $C - T$ is attached to the chords of D_ℓ for at most one ℓ. Thus, we observe that $\{C_1, \ldots, C_k\}$ is a partition of $C - c_{j+1}$ by Lemma 3.2.5 and the fact that C is connected (since therefore every indecomposable component of $C - T$ is attached to a chord of $T - c_{j+1}$); note that this also implicitly uses the fact that every neighbor of c_{j+1} is in T since the intersection order is an extension of the reachability order. We also have the following properties.

Claim 1. For all ℓ, (i) C_ℓ is connected and (ii) $t_1(C_\ell) > i_\ell$.

Proof. Note that clearly each neighbor of c_{j+1} attached to A_ℓ has a path to every other attached neighbor since each cross a chord in the outermost connected component of A_ℓ by definition. Furthermore, the traced subdiagram of T of a neighbor of c_{j+1} is connected since it is 1-terminal by Lemma 3.2.4. By construction, it follows that C_ℓ is connected since C is connected, giving (i). For (ii), note that either C_ℓ is constructed from the traced subdiagram of exactly one neighbor c_i of c_{j+1} or A_ℓ is nonempty. In the former case, c_i is the first terminal of C_ℓ, while in the latter case each chord of D_ℓ has a right neighbor in C_ℓ by the 1-terminality of traced subdiagrams. Thus, in either case, each neighbor of c_{j+1} in C_ℓ precedes (non-strictly) the first terminal chord of C_ℓ, which is necessarily either the neighbor c_i or in A_ℓ. Then another application of Lemma 3.2.4 yields the inequality $t_1(C_\ell) > i_\ell$. □

We now show that α has an inverse, that is, define a map β such that $\beta \circ \alpha = \mathrm{id}$ and $\alpha \circ \beta = \mathrm{id}$. Suppose we are given a tuple $((C_1, I_1), \ldots, (C_k, I_k))$ with $C_1, \ldots, C_k$ connected diagrams and $\{I_1, \ldots, I_k\}$ a partition of $[1, j]$ with $1 \leq |I_\ell| \leq t_1(C_\ell)$ for all ℓ and $j = |I_1| + \cdots + |I_k|$.

Definition 3.2.7. We define the composed diagram C and map β as follows.

- For all $1 \leq \ell \leq k$, D_ℓ: the 1-terminal diagram induced by the first $|I_\ell|$ chords of C_ℓ in the intersection order.
- $C^\emptyset$: the diagram obtained by concatenating $C_1, \ldots, C_k$ and then arranging the source-sink groups corresponding to the first $|I_1|, \ldots, |I_k|$ chords (in the intersection order) of $C_1, \ldots, C_k$, respectively, in order according to the permutation induced by $I_1, \ldots, I_k$, maintaining the order of the points within each diagram C_ℓ.
- C: the diagram obtained from $C^\emptyset$ by adding a chord c with its sink in the rightmost position and source immediately following the source-sink groups corresponding to the first j chords of $C^\emptyset$ (in the intersection order).
- $\beta((C_1, I_1), \ldots, (C_k, I_k)) = C$.

A representative example of this construction is given in Figure 3.10.

Claim 2. We have $t_1(C) = j + 1$.

Proof. For all ℓ, clearly at least one source of C_ℓ lies to the left of the source of c while the rightmost sink of C_ℓ lies to the right of the source of c. Since C_ℓ is connected, it follows

that c has a neighbor from each C_ℓ, implying that C is connected. Then by Lemma 1.3.3, c is the first terminal chord of C since its sink lies furthest to the right. Note that nonnesting paths of C_ℓ are preserved in C by the construction process; since the first terminal chord of C_ℓ clearly either crosses c or is nested under it, this implies that there is a nonnesting path from each of the first $|I_\ell|$ of C_ℓ to c, implying that $t_1(C) > |I_1| + \cdots + |I_\ell| + 1 = j + 1$ by Lemma 1.3.5. On the other hand, by the extension property of the intersection order, all the chords of $C_\ell - D_\ell$ come after the first terminal chord of C in the intersection order. Combined with the fact that only $|I_\ell|$ chords of D_ℓ have their sources to the left of the source of c, we see that $t_1(C) \leq j+1$ by Lemma 1.3.3. We therefore infer that $t_1(C) = j+1$, as required. □

Claim 3. The map β is the inverse of α.

Proof. Observe that α partitions the points of a given connected diagram C lying before the source of the terminal chord c_{j+1} along source-sink groups. It follows that, for C in the domain of α and $((C_1, I_1), \ldots, (C_k, I_k))$ in the domain of β, $\beta(\alpha(C)) = C$ and $\alpha(\beta((C_1, I_1), \ldots, (C_k, I_k)))$ by construction and our prior conclusions about α and β. □

The bijection α translates into the following result recursively enumerating connected diagrams by size and index of the first terminal chord.

Theorem 3.2.8. *Write $c_{n,j}$ for the number of connected diagrams of size n with $t_1(C) = j$. Then*

$$c_{n+1,j+1} = \sum_{k=1}^{n} \sum_{\substack{n_1 + \cdots + n_k = n \\ n_\ell > 1}} \sum_{\substack{i_1 + \cdots + i_k = j \\ 1 \leq i_\ell \leq n_\ell}} \binom{j}{i_1, \ldots, i_k} \sum_{j_1 > i_1} c_{n_1,j_1} \cdots \sum_{j_k > i_k} c_{n_k,j_k}.$$

This recurrence is implicit in the work of Marie and Yeats [75]. Later, an equivalent recurrence based on the root-share decomposition was obtained by Courtiel and Yeats [28]. While our decomposition is more complex, it also more clearly reveals the structure of connected diagrams and can be used to derive further information about certain additional diagram parameters, as shown in Chapter 4.

We now apply this decomposition to connected top-cycle-free diagrams. In this case the diagram ends up uniquely decomposing into a tuple of smaller top-cycle-free diagrams—the need to provide a shuffle-defining partition to accompany the decomposition effectively disappears. We specifically prove the following recurrence.

Theorem 3.2.9. *Write $t_{n,j}$ for the number of connected top-cycle-free diagrams of size n with $t_1(C) = j$. Then*

$$t_{n+1,j+1} = \sum_{k=1}^{n} \sum_{\substack{n_1+\cdots+n_k=n \\ n_\ell>1}} \sum_{\substack{i_1+\cdots+i_k=j \\ 1\le i_\ell\le n_\ell}} \sum_{j_1>i_1} t_{n_1,j_1} \cdots \sum_{j_k>i_k} t_{n_k,j_k}.$$

Proof. Let C be a connected top-cycle-free diagram of size $n+1$ with $t_1(C) = j+1$ and $c_1 \prec c_2 \prec \cdots \prec c_{n+1}$ the chords of C in the intersection order. As previously, let $\alpha(C) = ((C_1, I_1), \ldots, (C_k, I_k))$; Figure 3.11 displays a representative example. Note that $C_1, \ldots, C_k$ are clearly top-cycle-free since they are subdiagrams of the top-cycle-free diagram C.

Claim 4. $I_1, \ldots, I_k$ are intervals.

Proof. Suppose to the contrary that I_ℓ is not an interval for some ℓ. Then there exists a block $I_{\ell'}$ such that I_ℓ and $I_{\ell'}$ either nest or cross, that is, there are $r, s \in I_\ell$ and $t \in I_{\ell'}$ such that $r < t < s$. By construction, each subdiagram D_ℓ of C_ℓ contains a neighbor of the first terminal chord c_{j+1} of C and each such neighbor lies before the first terminal chord of C_ℓ in the intersection order. Since D_ℓ is connected, there exists a path P in D_ℓ from c_r to c_s. By construction of $D_{\ell'}$ and Lemma 1.3.5, there exists a nonnesting path Q in $D_{\ell'}$ from c_t to a neighbor of c_{j+1} in $D_{\ell'}$; let $Q' = Q \cup \{c_{j+1}\}$. Since $r < t < s$, by Lemma 1.3.3 $c_r < c_t < c_s$ in the standard order, implying that P and Q cross, that is, there is a chord c in P and a chord c' in Q such that c and c' cross. Let P' be a nonnesting path in D_ℓ from c to a neighbor of c_{j+1} in D_ℓ, which again exists by construction and Lemma 1.3.5. Then $P' \cup Q'$ contains two internally-disjoint nonnesting paths of length at least three with the same first and last chords (in the standard order), so one can easily see that it contains a top cycle as a subdiagram. But then C is not top-cycle-free, a contradiction. □

Now fix a tuple $((C_1, I_1), \ldots, (C_k, I_k))$ with $C_1, \ldots, C_k$ connected top-cycle-free diagrams and $\{I_1, \ldots, I_k\}$ a partition of $[1, j]$ into intervals with $1 \le |I_\ell| \le t_1(C_\ell)$ for all ℓ and $j = |I_1| + \cdots + |I_k|$, and write $C = \beta((C_1, I_1), \ldots, (C_k, I_k))$ and let c be the first terminal chord of C. Suppose C contains a top cycle subdiagram A. Clearly $C - c$ is top-cycle-free since its connected components are exactly $C_1, \ldots, C_k$. Then it follows that A must contain c and, since cycles are 2-connected, be a subdiagram of $C_\ell \cup \{c\}$ for some ℓ. Write $A = \{c'_1, \ldots, c'_m, c\}$, where $c'_1 < c'_2 < \cdots < c'_m < c$ in the standard order. Without loss of generality we may assume that A is chosen with c'_2 maximum (in the standard order). Then, since C_ℓ is top-cycle-free, there is no path nested under c'_2 from a chord of

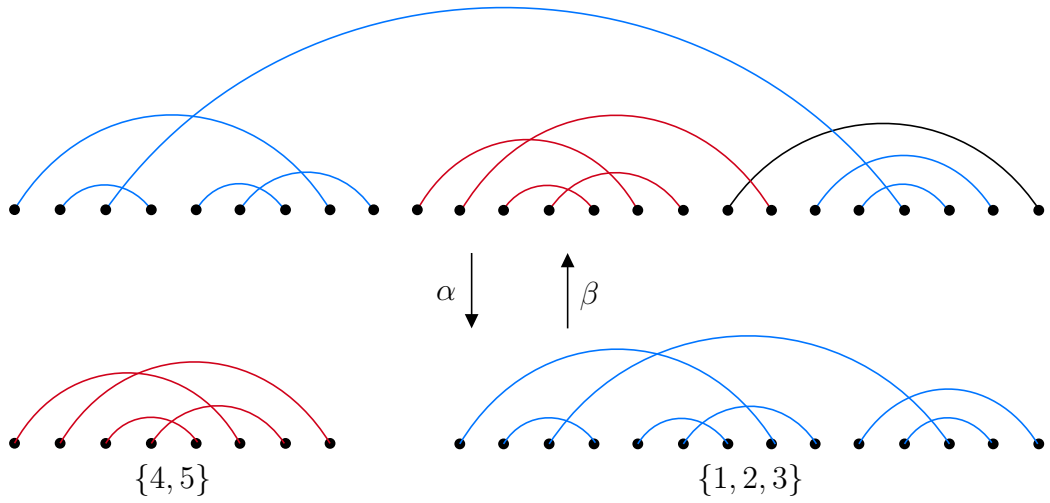
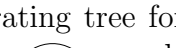
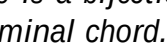


Figure 3.11: A top-cycle-free diagram and its decomposition defining the maps α and β.

$A^{\emptyset} = A - \{c_1^{\emptyset}, c_2^{\emptyset}, c\}$ to a right neighbor of t_2, implying that $A^{\emptyset}$ is a subset of the source-sink group of $c_2^{\emptyset}$. By construction it follows that $c_m^{\emptyset}$ cannot cross c, a contradiction. So C is top-cycle-free.

Since α is a bijection and there is exactly one partition of $[1, j]$ into a fixed number of intervals with specified ordered sizes, this completes the proof. □

In 2005, Jelínek [58] showed that top-cycle-free diagrams are in bijection with the classes $\mathcal{D}($  $)$ and $\mathcal{D}($  $)$. He proved this by constructing a somewhat complicated generating tree for each of these classes and showing that the two trees are isomorphic. Since  and its reverse are connected, this result also implies that the connected subsets of these classes are all equinumerous. Here, we show that the decomposition map α gives a recursively-defined direct bijection between $\mathcal{C}_n(T_{>3})$ and $\mathcal{C}_n($  $)$. The bijection preserves both size and index of the first terminal chord, in virtue of the common decompositions relating a diagram and its image. Note that the fact that this bijection preserves the index of the first terminal chord was not obtained in [58].

Theorem 3.2.10. *There is a bijection between $\mathcal{C}(T_{>3})$ and $\mathcal{C}($  $)$ that preserves size and index of the first terminal chord.*

Proof. Let C be a  -free diagram of size n with $t_1(C) = j$ and $c_1 \prec c_2 \prec \cdots \prec c_n$ the chords of C in the intersection order. Write $\alpha(C) = ((C_1, I_1), \ldots, (C_k, I_k))$, and carry over all prior associated notation appropriately. See Figure 3.12 for a representative example

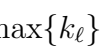
of α applied to  -free diagram. As previously, note that $C_1, \ldots, C_k$ are -free since they are subdiagrams of C. The key claim is that the blocks $I_1, \ldots, I_k$ are uniquely determined by their sizes $i_1, \ldots, i_k$ and the subdiagrams $C_1, \ldots, C_k$ in the following sense.

Claim 5. Each block I_ℓ partitions into intervals $I_{\ell,1} > \ldots > I_{\ell,k_\ell}$ with the following properties.

1. For all $1 \leq r < s \leq j-1$, we have $r, s \in I_{\ell,i}$ if and only if
   a) $i = 1$ and the sinks of c_r and c_s both lie in A_ℓ,
   b) $i > 1$ and the sinks of c_r and c_s both lie in source-sink groups associated to $I_{\ell,i-1}$, or
   c) $\ell = 1$, $i = k_1$, $k_1 = \max\{k_\ell\}$, and the sink of c_r lies in a source-sink group associated to I_{1,k_1}.
2. The intervals interleave, that is, $I_{k,i} > I_{k-1,i} > \cdots > I_{1,i} > I_{k,i+1}$.

Note that there may be fewer intervals in some blocks than others, so not all terms in the inequalities implied by (2) are necessarily present. It is for this reason the intervals are labeled in reverse of the natural order.

Proof of Claim 5. Note that if $k = 1$ the claim reduces to the trivial statement that I_1 is an interval, so we may assume $k > 2$. We prove both properties simultaneously by induction on i, the secondary interval index.

By  -avoidance, the neighbors of the terminal chord c_j of C form a clique. Thus all such neighbors in C_ℓ cross all such neighbors in $C_{\ell'}$, implying that the sources of $C_\ell \cap N_C(c_{j+1})$ lie rightward of the sources of $C_{\ell'} \cap N_C(c_{j+1})$ for $\ell' < \ell$. This gives all but possibly the last inequality of (2) and gives the "if" direction of (1): $r, s \in I_{\ell,1}$ if the sinks of c_r and c_s lie in A_ℓ. Now, if the other direction of (1) does not hold, then there exists $1 < t < j-1$ such that $c_t \in I_{\ell,1}$ and the sink of c_t lies in the source-sink group of a neighbor c_r of the terminal chord c_j. Observe that $\ell = 1$ since otherwise c_r, c_t, and a neighbor of c_j contained in $C_{\ell-1}$ (which necessarily exists since $k > 2$) form a subdiagram isomorphic to  . Furthermore, by contradiction, we are not in case (c), so we may assume that $1 \neq \max\{k_\ell\}$. Consequently there must be $\ell' > \ell$ such that $C_{\ell'}$ has a chord c whose sink lies in a source-sink group associated to $I_{\ell',1}$ and source lies left of the source of c_t. But this implies that c_r, c_t, and c form a subdiagram isomorphic to  , yielding a contradiction. For the base case it only remains to show for (2) that $I_{k,2} < I_{1,1}$ when $k_k > 2$, but this follows from

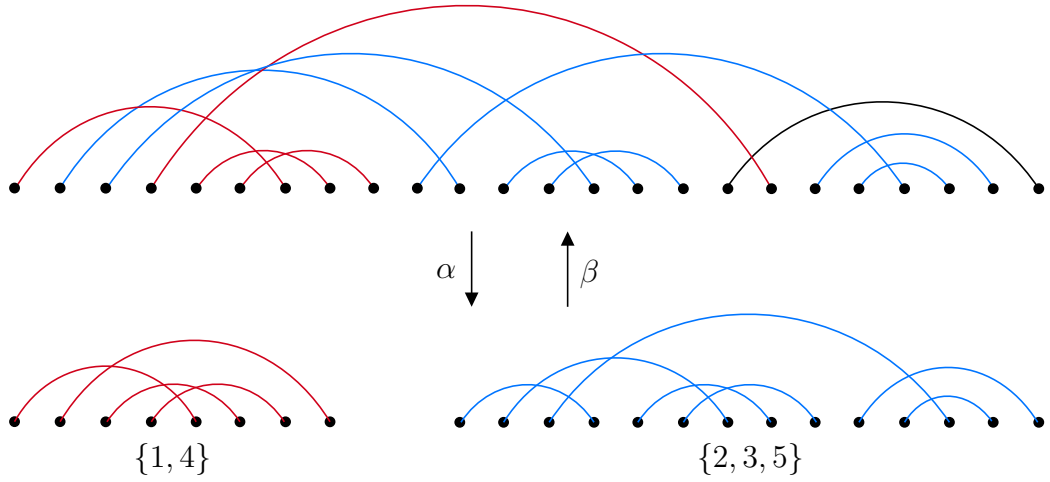


Figure 3.12: A  -free diagram and its decomposition defining the maps α and β.

the fact that otherwise there exist $c \in C_1 \cap N_C(c_j)$, $c' \in C_k \cap N_C(c_j)$, and a left neighbor c'' of c' such that $\{c < c'' < c'\}$ is isomorphic to  .

Now suppose $i > 1$ and assume (1) and (2) hold for all smaller i. For $r, s \in I_{\ell,i}$, observe that if we are not in case (c) then the sinks of c_r and c_s must lie in source-sink groups associated to $I_{\ell,i}$ or $I_{\ell,i\ -1}$; otherwise the sink of c_r or c_s lies in a source-sink group associated to $I_{\ell,i'}$ for $i' < i - 1$, violating the induction hypothesis for $i' + 1$. If one of the first $k - 1$ inequalities of (2) does not hold, then there exists $\ell > \ell'$ such that $I_{\ell,i} < I_{\ell',i}$. Then by the induction hypothesis, connectivity, and our above observation there exist chords c_t, c_r, and c_s such that (i) $t \in I_{\ell,i}$ and the sink of c_t lies a source-sink group associated to $I_{\ell,i\ -1}$, (ii) $s \in I_{\ell',i\ -1}$ and the sink of c_s lies to the right of the sink of c_t, and (iii) $r \in I_{\ell',i}$ and c_r is attached to the source-sink group of c_s. Then these three chords form a subdiagram isomorphic to  , a contradiction. A near identical argument further yields the last inequality of (2).

Now we turn to property (1). For the "if" direction, the statement is trivial for case (c), so we may assume that we are in case (b). Furthermore, we may assume by the induction hypothesis and by way of contradiction that $s \in I_{\ell,i}$ and $r \in I_{\ell,i'}$ for $i' > i$. Then by property (2) there exists $\ell' < \ell$ such that $I_{\ell',i} < I_{\ell,i} < I_{\ell',i\ -1} < I_{\ell,i\ -1}$. It follows that, by connectivity, there exists a chord c in $C_{\ell'}$ such that the source of c lies in between the source of c_s and the source c_r, and the sink of c lies in a source-sink group of a chord c' that crosses c_r. Consequently, $\{c_r < c < c'\}$ is isomorphic to  . The "only if" direction proceeds sufficiently similar to the base case that we omit it. □

Since A_ℓ is obtained by removing the first i_ℓ chords of C_ℓ in the intersection order, it is easy to see that the blocks are indeed inductively uniquely determined by their sizes and the subdiagrams $C_1, \ldots, C_k$. We thus construct a bijection $\rho\colon \mathcal{C}_n(\overset{\frown}{\cdots}) \to \mathcal{C}_n(T_{>3})$ as follows:

- obtain the recursively-defined images $\hat{C}_1 = \rho(C_1)$, $\ldots$, $\hat{C}_k = \rho(C_k)$,
- let $\hat{I}_1, \ldots, \hat{I}_k$ be the intervals of size $i_1, \ldots, i_k$, and
- set $\rho(C) = \beta((\hat{C}_1, \hat{I}_1), \ldots, (\hat{C}_k, \hat{I}_1))$.

Figure 3.11 shows the image of the diagram from Figure 3.12 under this bijection. By the construction of β this bijection preserves size and index of the first terminal chord. □

3.2.3 Triangulations, maps, and other friends

While Theorem 3.2.9 in principle provides sufficient information to obtain an explicit formula for the counting sequence of $\mathcal{C}(T_{>3})$, the nested composition-indexed series contained in the recurrence makes this a difficult task in practice. For this we turn to instead applying a bijective approach, for which our decomposition will prove helpful. Along the way we explore several connections between top-cycle-free diagrams and other natural combinatorial objects.

Marie and Yeats implicitly obtained Theorem 3.2.8 by passing from chord diagrams to rooted plane binary trees. This essentially amounts to showing that there is a recursively-defined bijection, based on the root-share decomposition, between rooted connected chord diagrams and rooted plane binary trees with two recursively-defined properties. These properties are somewhat technical and based on a labeling of the leaves induced via the bijection by the intersection order on the chords of the associated diagram.

Later, Courtiel, Yeats, and Zeilberger [30] discovered a much nicer type of object, combinatorial maps, that could substitute for connected chord diagrams in their work. A *(rooted) combinatorial map* is a transitive permutation representation of the group $\langle \sigma, \alpha \mid \alpha^2 = 1 \rangle$ with a distinguished fixed point for the action of α (the root). It can be represented by a connected graph of half edges, each paired with at most one other half-edge to form an edge, with a cyclic order of the half-edges incident to each vertex and the root a vertex attached to a distinguished "dangling" half-edge. Such a map inherits certain properties of this underlying graph; in particular, it is *bridgeless* if its graph is bridgeless, that is, 2-edge-connected. Other properties require a map-specific definition: it is *planar* if its Euler characteristic is 2 (see e.g. [30] for the definition of the Euler characteristic).

Theorem 3.2.11 (Courtiel, Yeats, and Zeilberger [30]). *There exists a bijection θ between connected diagrams and bridgeless maps such that*

- *chords correspond to edges;*
- *terminal chords correspond to vertices;*
- *the position t_1 of the first terminal chord in the intersection order corresponds to the indegree $\deg_{DFS,1}$ of the root vertex v_1 under the orientation induced by the rightmost depth-first search.*

For brevity, we omit a definition of the rightmost depth-first search and associated orientation referenced in this theorem; see Section 5.5 of [30] for these details.

It turns out top-cycle-free diagrams also independently arise in the work of Courtiel, Yeats, and Zeilberger as the image of the bijection θ on a natural subset of bridgeless maps: those which are planar.

Theorem 3.2.12 (Courtiel, Yeats, and Zeilberger [30, Proposition 27]). *Under θ, planar bridgeless maps are in bijection with connected top-cycle-free diagrams.*

Furthermore, changing "connected" to "indecomposable" corresponds to dropping "bridgeless" for both this result and Theorem 3.2.11.

Top-cycle-free diagrams turn out to be related to an even more classic type of combinatorial object: triangulations. In this context, a ***triangulation*** T is a plane graph in which every bounded face is a triangle; for technical reasons we also require that a single edge counts as a triangulation. Triangulation T is ***rooted*** if it has a distinguished edge, the ***root***; we will only work with rooted triangulations. As is standard in the literature, we consider such triangulations up to root-preserving isomorphism. ***Exterior vertices*** of T are incident to its boundary face, while all other vertices of T are ***interior vertices***. We obtain the following result, whose proof follows shortly.

Theorem 3.2.13. *There is a bijection ω between (a) connected top-cycle-free diagrams with n chords and $t_1 = i$ and (b) triangulations with $n - i$ interior vertices and $i + 1$ exterior vertices.*

In 1964, William G. Brown [16] explicitly enumerated (rooted) triangulations with n interior vertices and $m + 3$ exterior vertices by deriving and solving a functional equation for the associated bivariate ordinary generating function, showing their number to be

$$\frac{2(2m+3)!(4n+2m+1)!}{m!(m+2)!n!(3n+2m+3)!}.$$

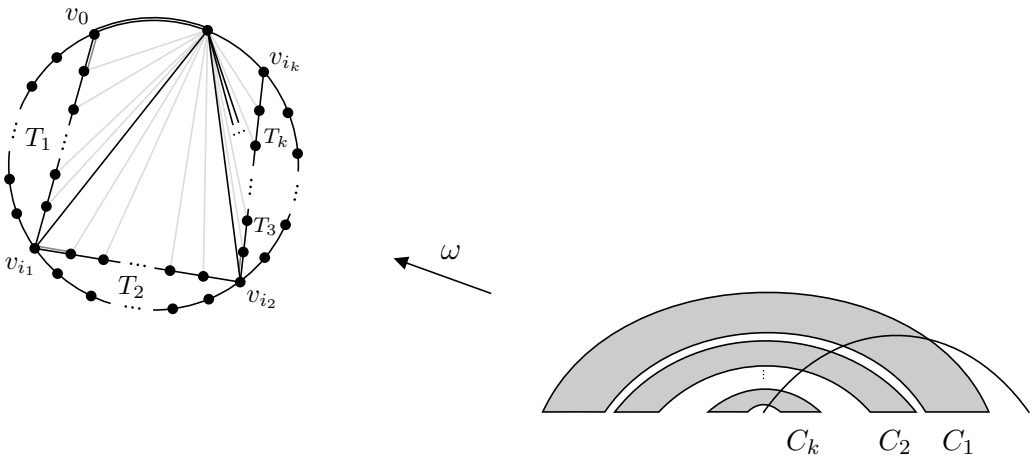
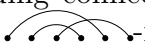


Figure 3.13: The recursive construction of a triangulation and its image chord diagram under the bijection ω. The exterior vertex labels are taken from the decomposition map γ.

From Theorems 3.2.12, 3.2.13, and 3.2.10 it follows that we get an explicit count for the corresponding connected top-cycle-free diagrams and planar bridgeless maps, as well as connected  -free diagrams.

Corollary 3.2.14. *The number of both (a) connected top-cycle-free diagrams with n chords and $t_1 = i$, (b) connected  -free diagrams with n chords and $t_1 = i$, and (c) planar bridgeless maps with n edges and $\deg_{DFS,1} = i$ is*

$$\frac{2(2i-1)!(4n-2i-3)!}{(i-2)!i!(n-i)!(3n-i-1)!} = \frac{i-1}{(4n-2i-1)(2n-i-1)}\binom{2i-1}{i}\binom{4n-2i-1}{n-i}.$$

In particular, for $n > 1$,

$$|\mathcal{C}_{n+1}(T_{>3})| = \frac{2}{n(n+1)}\binom{4n+1}{n-1}. \tag{3.2}$$

Proof of Theorem 3.2.13. We will use the unique decomposition described in the previous section to recursively define our bijection. There is exactly one (connected top-cycle-free) diagram of size 1, namely, a single chord, and we map it to the unique (rooted) triangulation with no interior vertices and 2 exterior vertices: a single edge. So, explicitly, ω maps a single chord to a single edge. Now fix a connected top-cycle-free diagram C of size $n > 2$. Applying the unique decomposition from previous sections, we get connected

top-cycle-free subdiagrams $C_1, \ldots, C_k$ and integers $i_1, \ldots, i_k$ with $i_1 + \cdots + i_k = t_1(C) - 1$ and $1 \leq i_\ell \leq t_1(C_\ell)$ for all $1 \leq \ell \leq k$; we have skipped the intervals $I_1, \ldots, I_k$ because their lengths $i_1, \ldots, i_k$ are sufficient to specify the decomposition and define the bijection ω. Then inductively we obtain image triangulations $T_1, \ldots, T_k$ of $C_1, \ldots, C_k$ under ω; write $t_\ell = t_1(C_\ell)$ and $v_{\ell,0}, \ldots, v_{\ell,t_\ell}$ for the exterior vertices of T_ℓ read counterclockwise starting at the leftmost vertex $v_{\ell,0}$ of the root edge (so, in particular, $v_{\ell,0}v_{\ell,t_\ell}$ is the root edge). Then we construct triangulation T as follows:

1. Join $T_1, \ldots, T_k$ in that order by identifying $v_{\ell\ 1,i_\ell}$ and $v_{\ell,0}$.
2. Add a new vertex v and connect it with an edge to $v_{1,0}$ and v_{k,i_k}; this creates a single new bounded face.
3. Root the graph at edge $v_{1,0}v$.
4. While keeping the graph simple, add an edge from v to every vertex incident to the newly created bounded face.

See Figure 3.13 for a visual representation of the construction. We then set $\omega(C) = T$. It is easy to see that T is indeed a triangulation. Furthermore, the number of vertices of T is

$$|T_1| + \sum_{\ell=2}^{k} (|T_\ell| - 1) = |C_1| + 1 + \sum_{\ell=2}^{k} |C_\ell| = |C|$$

while the number of external vertices of T is

$$1 + \sum_{\ell=1}^{k} i_\ell + 1 = t_1(C) + 1.$$

We have thus defined ω; in particular, if we write δ for the function sending $((T_1, i_1), \ldots, (T_k, i_k))$ to T then $\omega = \delta \circ \omega \circ \alpha$ (where ω acts on a tuple of diagram-integer pairs in the obvious way). It remains only to show that ω is a bijection. To that purpose, it suffices to define a unique decomposition γ of a triangulation T reversing the above construction, that is, inverting δ. This follows from the observation that

$$\beta \circ \omega^{-1} \circ \gamma = \alpha^{-1} \circ \omega^{-1} \circ \delta^{-1} = (\delta \circ \omega \circ \alpha)^{-1} = \omega^{-1},$$

so we automatically get a recursively-defined inverse of ω. So, let T be a triangulation and $v_0, v_1, \ldots, v_i$ be the exterior vertices of T read counterclockwise with v_0v_i the

root edge. Again reading counterclockwise, vertex v_i has two or more external neighbors $v_0, v_{i_1}, v_{i_2}, \ldots, v_{i_{k-1}}, v_{i_k}$. Then deleting all edges incident to v_i leaves a sequence of unrooted triangulations $T_1, \ldots, T_k$ pairwise joined at vertices $v_{i_1}, \ldots, v_{i_{k-1}}$ (see Figure 3.13). Root each T_ℓ at the unique edge incident to $v_{i_{\ell-1}}$ which was previously incident to a bounded face, where $i_0 = 0$. Then it readily follows that setting $\gamma(T) = ((T_1, i_1), \ldots, (T_k, i_k))$ gives the desired decomposition; T_ℓ has exactly $i_\ell + 1$ exterior vertices and, furthermore, clearly $\delta(\gamma(T)) = T$. This concludes the proof. □

Tutte [102] was the first to prove that triangulations with n internal vertices and 3 external vertices are counted by (3.2), while Wash and Lehman [106] proved that this also counts the number of rooted bridgeless planar maps with n edges. Bijections connecting these two sets of objects were later given by Wormald [109], Fusy [49], and Fang [41], with the former two obtained recursively and the latter directly. Both our bijection between connected top-cycle-free diagrams and triangulations and the bijection of Courtiel, Yeats, and Zeilberger between planar bridgeless maps and connected top-cycle-free diagrams are recursively-defined. Obtaining a direct bijection mapping connected top-cycle-free diagrams to these other objects remains open.

There are more than a few other well studied objects in bijection with triangulations and, thereby, connected top-cycle-free diagrams. Tamari [100] defined a partial order on the set of plane binary trees with n non-leaf vertices, which are counted by the Catalan numbers, and proved that it specified a lattice, the n^{th} Tamari lattice $\mathcal{L}_n^T$, whose covering relation is defined by right rotation of subtrees. Let $\text{Int}(P)$ denote the set of all intervals of a poset P. Chapoton [22] proved that the cardinality of $\text{Int}(\mathcal{L}_n^T)$ is given by (3.2); later, Bernardi and Bonichon [6] provided a bijection between Tamari intervals and triangulations explaining their common count. Along with the bijection between triangulations and bridgeless planar maps, Fang [41] obtained a bijection between bridgeless planar maps and Tamari intervals; all of these bijections passed through so-called "sticky trees".

In 2020, Colin Defant [35] proved that Tamari intervals are in bijection with uniquely-sorted permutations[5] of length $2n+1$ that avoid the patterns 132 or 231. This is analogous to the classic result of Knuth that 132- and 231-avoiding permutations of $\mathcal{S}_n$ are counted by the n^{th} Catalan number $C_n = \frac{1}{n+1}\binom{2n}{n}$.

While we will return to these objects later when we consider other chord diagram classes, for now we summarize these relations in the following corollary.

Corollary 3.2.15. *Connected top-cycle-free diagrams of size $n+1$ are equinumerous with*

[5] Recall these permutations from the start of the chapter.

intervals of the n^{th} Tamari lattice and $(132, 231)$-avoiding uniquely sorted permutations of length $2n + 1$.

Along with Equation (3.2) this makes it clear that the class $\mathcal{C}(T_{>3})$ is another example of the offset phenomenon.

3.3 The map $\psi\colon \mathcal{T}_{n+1} \to \mathcal{D}_n$

As mentioned previously, Courtiel and Yeats [28] obtained a recurrence relation for the number $c^{\emptyset}_{n,k}$ of connected diagrams with n chords such that $t_1 > n - k$. In addition to recursively computing the associated exponential generating function and estimating the asymptotics of $c^{\emptyset}_{n,k}$, they used this result to derive the following count.

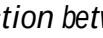
Theorem 3.3.1 (Courtiel and Yeats [28, Corollary 11])**.** *The number of 1-terminal diagrams of size $n + 1$ is $(2n - 1)!!$.*

This is the same as the number of (arbitrary) diagrams of size n. Courtiel and Yeats used the root-share decomposition to inductively prove this count. In this section we describe and study a bijection between the set $\mathcal{T}_{n+1}$ of 1-terminal diagrams with $n + 1$ chords and the set $\mathcal{D}_n$ of diagrams with n chords. This map was actually first discovered by Yeats almost a decade ago in the course of the original work with Marie [75] on chord diagram generating series solutions to Dyson-Schwinger equations, but never published or, to our knowledge, significantly studied.[6] Yeats used a recursive formulation related to the inductive proof of Theorem 3.3.1; here we describe and concentrate on a new, simpler formulation. We will briefly return to Yeats' formulation in Section 3.3.2 in the context of Stirling permutations.

Theorem 3.3.2. *There is a bijection $\psi\colon \mathcal{T}_{n+1} \to \mathcal{D}_n$ between 1-terminal diagrams of size $n + 1$ and diagrams of size n. Furthermore, the bijection*

(i) preserves nestings,

(ii) reduces the number of crossings by n,

(iii) restricts to a bijection between connected nonnesting diagrams $\mathcal{C}_{n+1}($  $)$ and nonnesting diagrams $\mathcal{D}_n($  $)$, and

[6] Personal communication.

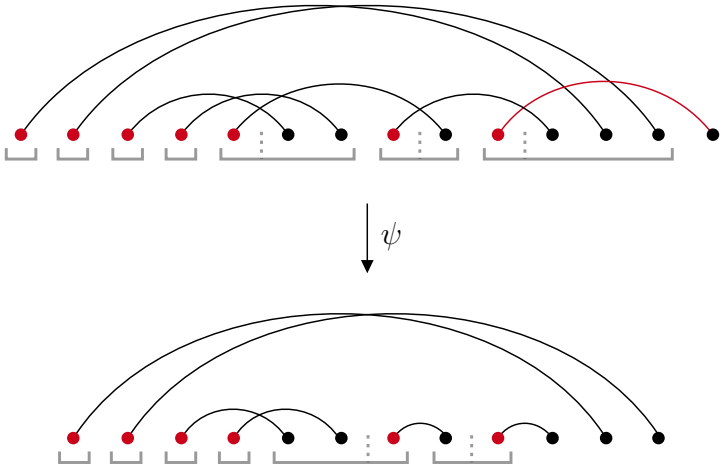


Figure 3.14: An example of a 1-terminal diagram and its image diagram under the bijection ψ. The source-sink groups "flipped" into sink-source groups by ψ are indicated by the horizontal brackets, with the "flip axes" indicated by the dotted lines, and the sources and terminal chord of T are indicated in red.

(iv) restricts to a bijection between 1-terminal top-cycle-free diagrams $\mathcal{T}_{n+1}(T_{>3}) = \mathcal{T}_{n+1}(T_{>3}, B_{>3})$ and noncrossing diagrams $\mathcal{D}_n(\frown)$.

Proof. Consider $T \in \mathcal{T}_{n+1}$ with chords $c_1 < c_2 < \cdots < c_{n+1}$ in the standard order (which agrees with the intersection order by 1-terminality). All but the last point of T partition into a sequence of source-sink groups, one for each source of T. Recall that, since T is 1-terminal, these groups only contain the source and maximal interval of sinks immediately following the source. Then we define $\psi(T)$ to be the diagram obtained by

(1) moving each source to the end of its source-sink group, and

(2) deleting the formerly terminal chord c_n.

Figure 3.14 displays a representative example of T and $\psi(T)$. Note that the bijection ψ induces a mapping, given by the construction, between the chords of T and the chords of its image $\psi(T)$. We can therefore abuse notation and write $\psi(c)$ for the image of a non-terminal chord c under this induced mapping.

Step (1) converts all but the last source-sink group in T into a corresponding *sink-source group* in $\psi(T)$, consisting of a source together with the maximal interval of sinks adjacent to the source on the left. Observe that we can also dually characterize this as

uncrossing each non-terminal chord c with its rightmost right neighbor $c^{\emptyset}$ by moving the source of $c^{\emptyset}$ immediately ahead of the sink of c while maintaining the relative order of all other endpoints. In particular, the following key property holds.

Claim 6. A non-terminal chord $c \in T$ has k right neighbors if and only if $\psi(c)$ has $k-1$ right neighbors.

This gives property (ii). Observe that for non-terminal chords $c, c^{\emptyset} \in T$, $c^{\emptyset}$ is nested under c if and only if $\psi(c^{\emptyset})$ is nested under $\psi(c)$ by the construction of ψ. Since the terminal chord is necessarily not part of any nestings we infer property (i).

Now clearly the codomain of ψ is $\mathcal{D}_n$, so to prove that ψ is a bijection between the designated sets it suffices to exhibit its inverse. To that end, let $D \in \mathcal{D}_n$ and define $\phi(D)$ to be the diagram obtained by

($1^{\emptyset}$) concatenating a single chord to the end of D, and

($2^{\emptyset}$) moving the source of each sink-source group to the beginning of its sink-source group.

Note that we could also skip using ($2^{\emptyset}$) on the terminal chord and instead of ($1^{\emptyset}$) directly add a chord covering the rightmost maximal interval of sinks. It is readily apparent that ($1^{\emptyset}$) and ($2^{\emptyset}$) invert (2) and (1), respectively, implying that ϕ is the inverse of ψ.

It remains only to prove properties (iii) and (iv). For the former, Corollary 1.3.6 implies that connected nonnesting diagrams are 1-terminal, so the property follows by nesting preservation. For the latter, property (iv), we require a structural characterization of 1-terminal top-cycle-free diagrams. Recall that T is 1-terminal.

Claim 7. Diagram T is top-cycle-free if and only if T is a tree[7] and every non-terminal chord has exactly one right neighbor.

Proof. The "if" direction trivially holds. For the other direction, assume T is top-cycle-free. Note that bottom cycle diagrams have chords with two right neighbors, so it suffices to show that every non-terminal chord has exactly one right neighbor (since T consequently must be a tree). If not, then there exists non-terminal $c \in \mathcal{T}$ with at least two right neighbors d and $d^{\emptyset}$. Since T is top-cycle-free d and $d^{\emptyset}$ do not cross and we may assume that d is nested under $d^{\emptyset}$. Choose e as far right as possible such that there is a nonnesting d-e path nested under $d^{\emptyset}$. Then either e has a right neighbor crossing $d^{\emptyset}$, in which case T contains a top cycle subdiagram, or e is terminal. In either case we get a contradiction. □

[7] That is, $G(T)$ is a tree.

Note that we could also obtain this claim using a quicker induction argument, but the above proof offers more insight. It in particular reflects the fact that bottom cycle diagrams are not 1-terminal.

To obtain property (iv), suppose T is top-cycle-free. Then it follows by Claims 7 and 6 that each chord in the image $\psi(T)$ has no right neighbors, implying that there are no crossings at all. We can similarly infer that for each noncrossing diagram D every non-terminal chord in $\psi^{-1}(D)$ has exactly one right neighbor, as required. □

Among other things, this sheds further light on the consequence of Theorem 3.2.13 that 1-terminal top-cycle-free diagrams are in bijection with triangulations with no interior vertices and, therefore, are counted by the Catalan numbers. In a nutshell, (iv) implies that we can think of 1-terminal top-cycle-free diagrams as the connectivity 1 equivalent to noncrossing diagrams, since trees have connectivity 1. As indicated by Claim 7, they are also minimally 1-terminal in the sense that each chord has the minimum number of right neighbors required to be 1-terminal.

3.3.1 Higher terminality

In this section, we study the relation between k-terminality and the bijection ψ. With the observations at the end of the previous section in mind, we can generalize statement (iv) of Theorem 3.3.2 to get a connectivity k equivalent to noncrossing diagrams. Call a diagram C *k-terminal-minimal* if all but its last $k-1$ chords have exactly k right neighbors. Clearly C is k-terminal and furthermore, in particular, 1-terminal top-cycle-free diagrams are 1-terminal-minimal. We will require a basic fact about connectivity, followed by two important statements about the relationship between k-connectivity, k-terminality, and ψ. For a simple graph G, write $G[A]$ for the induced subgraph on $A \subseteq V(G)$ and $\delta_{G\ A}(A)$ for the minimum number of neighbors that a vertex in A has in $G - A$.

Lemma 3.3.3. *If $G - A$ is k-connected, $G[A]$ is either k-connected or has size at most k, $\delta_A(V(G) - A) > \min\{k, |A|\}$, and $\delta_{G\ A}(A) > k$, then G is k-connected.*

We omit the straightforward proof of this fact.

Proposition 3.3.4. *If diagram C is k-terminal and has at least $k+1$ chords then it is k-connected.*

Proof. Let c be the root chord of a k-terminal diagram C of size at least $k+1$. By definition $C - c$ is k-terminal, so it is either the complete diagram of size k or we may

inductively assume that it is k-connected. In the former case C is also complete and thus k-connected, while in the latter case the fact that c has k right neighbors in C implies that C is k-connected by Lemma 3.3.3 with $A = \{c\}$. □

Proposition 3.3.5. *The map ψ restricts to a bijection between k-terminal diagrams of size n and $(k-1)$-terminal diagrams of size $n-1$.*

Proof. As with (iv) of Theorem 3.3.2, this is a straightforward consequence of Claim 6. □

We similarly get our desired conclusion.

Proposition 3.3.6. *The map ψ restricts to a bijection between k-terminal-minimal diagrams of size n and $(k-1)$-terminal-minimal diagrams of size $n-1$.*

Corollary 3.3.7. *The map ψ^k on k-terminal diagrams restricts to a bijection between k-terminal-minimal diagrams of size $n+k$ and noncrossing diagrams of size n. In particular, k-terminal-minimal diagrams are counted by the Catalan numbers.*

While Claim 7 provides a structural characterization of 1-terminal-minimality, we have yet to obtain such a characterization for k-terminal-minimality. Our preliminary investigations indicate though that there should be a similar description in terms of k-terminal diagrams forbidding an infinite class of subdiagrams.

Conjecture 3.3.8. *There is a "nice" infinite forbidden subdiagram characterization of k-terminal-minimality.*

In his infamous compilation of combinatorial objects counted by the Catalan numbers, Stanley [96] used ballot sequences and functions $f : \mathbb{N} \to \mathbb{N}$ satisfying $f(i) \leq i$ for all i to define $2^{\aleph_0}$ chord diagram interpretations of the Catalan numbers $\{C_n\}_{n>0}$. The noncrossing and nonnesting diagrams are recovered by setting $f(i) = 1$ and $f(i) = i$, respectively, so in a sense these interpretations lie between the two. Yet this construction only gives a finite number of interpretations for any fixed size n. With the set of k-terminal-minimal diagrams we have provided a countable infinity of combinatorial interpretations of C_n. Furthermore, combining Stanley's construction together with the map ψ gives $2^{\aleph_0}$ combinatorial interpretations of the Catalan numbers with $\aleph_0$ interpretations at each fixed n.

We now turn to generalizing statement (iii) of Theorem 3.3.2. The proof used the fact that connectivity is equivalent to 1-terminality for nonnesting diagrams. This actually points to a more general statement about nonnesting diagrams.

Lemma 3.3.9. *A nonnesting diagram is k-connected if and only if it is k-terminal and has at least k chords.*

Proof. By Proposition 3.3.4 it suffices to prove the "only if" direction. Let C be nonnesting and k-connected and c be its root chord. By the Erdös-Szekeres theorem the left and right neighborhoods of each chord of C form cliques, implying by k-connectivity that the neighborhood of c is a clique of size at least k. It follows that c cannot be part of any minimal vertex cut, so $C - c$ is either complete or k-connected. In the former case we are done, while in the latter case we inductively get that $C - c$ is k-terminal, so C is as well. □

Combining this observation with Theorem 3.3.2 and Proposition 3.3.5, we get the following.

Proposition 3.3.10. *The map ψ restricts to a bijection between k-connected nonnesting diagrams of size n and $(k-1)$-connected nonnesting diagrams of size $n-1$.*

Corollary 3.3.11. *The map ψ^k restricts to a bijection between k-connected nonnesting diagrams of size $n+k$ and nonnesting diagrams of size n. In particular, k-connected nonnesting diagrams are counted by the Catalan numbers.*

3.3.2 Relationship with other double factorial objects

Recall that there are a number of other classical combinatorial objects counted by the double factorials $(2n-1)!!$ in addition to chord diagrams. We will briefly focus on two of the most notable: increasing ordered trees and Stirling permutations. A (rooted) tree T of size n is *ordered* or *plane* if it is equipped with a total ordering of the children of each vertex $v \in T$. The tree T is *increasing* if its vertices are labelled with the integers $0, 1, 2, \ldots, n-1$ such that the label of any child is greater than its parent. A *Stirling permutation* of size n is a permutation of the multiset $\{1, 1, 2, 2, \ldots, n, n\}$ such that, for all i, all values between the two occurrences of i are greater than i. Stirling permutations were introduced by Gessel and Stanley to give a combinatorial interpretation to the Stirling polynomials. They have since been studied extensively and generalized in multiple different ways (see e.g. [50, 12, 56, 57]), as have increasing ordered trees (see e.g. [4, 83]). Write $\mathcal{S}_n$ and $\mathcal{I}_n$ for the sets of Stirling permutations and increasing ordered trees of size n, respectively.

There are classic recursive constructions of both $\mathcal{C}_n$ and $\mathcal{S}_n$: insert a root chord and insert nn into each element of $\mathcal{C}_{n-1}$ and $\mathcal{S}_{n-1}$, respectively, in all possible ways. Since

there are always $2n-1$ possible insertion places, this proves that both of these sets have cardinality $(2n-1)!!$ and gives a recursively-defined bijection ζ between them. This leads into a simple characterization of 1-terminality and the bijection ψ on Stirling permutations.

Proposition 3.3.12. *Diagram C is 1-terminal if and only if $\zeta(C)$ has 1 as a pre x and su x. In this case, $\zeta(\psi(C))$ is obtained from $\zeta(C)$ by removing both occurrences of 1 and normalizing the alphabet.*

Proof. Suppose $\zeta(C)$ does not have 1 as a prefix or suffix; by possibly taking the reverse, we may assume the former. Then, since 11 clearly has 1 has a prefix, at some step in the iterative construction of C a root chord is concatenated to the front of the currently placed chords, implying that it must be terminal in C and not the terminal chord with greatest sink. So C is not 1-terminal. On the other hand, if C is 1-terminal, this never occurs, so it easily follows that 1 remains a prefix and suffix whenever a new element is placed and, therefore, is a prefix and suffix of $\zeta(C)$; this gives the first statement. For the second statement, let $C^{\emptyset}$ be obtained by removing the root chord c of C. Observe that $\psi(C^{\emptyset}) = \psi(C) - \psi(c)$ and $\zeta(C^{\emptyset}) = \zeta(C) - nn$. Then the result follows straightforwardly from induction. □

The classic bijection η between increasing ordered trees and Stirling permutations sends trees of size $n+1$ to permutations of size n: for a tree $T \in \mathcal{I}_{n+1}$, delete the root label and transfer the remaining labels from vertices to parent edges, then take a pre-order traversal around the tree, traversing each edge twice. The encountered sequence of labels is the Stirling permutation $\eta(T)$.

As hinted at prior in the paper, and from the fact that there are $(2n-1)!!$ increasing ordered trees of size $n+1$, the decomposition from Section 3.2.2 defined by the maps α and β recursively defines a natural bijection θ between the set $\mathcal{T}_{n+1}$ of 1-terminal diagrams and the set $\mathcal{I}_{n+1}$ of increasing ordered trees. For each $C \in T_{n+1}$, apply α to get the 1-terminal subdiagrams $C_1, \ldots, C_k$ and intervals $I_1, \ldots, I_k$ of $[1, n]$. Then recursively apply θ to $C_1, \ldots, C_k$ to get increasing ordered trees $T_1, \ldots, T_k$, graft them to a new root of label 0 in that order, and finally apply an order-preserving bijection to reassign each subtree T_ℓ with the labels from I_ℓ. The resulting increasing ordered tree T is then set as the image of C under θ. We omit the proof that this actually defines a bijection between $\mathcal{T}_{n+1}$ and $\mathcal{I}_{n+1}$; it proceeds similarly to the proof of Theorem 3.2.13. Then, in this context, ψ plays the role of extending this to a bijection to chord diagrams of size n.

From all of the above we get two bijections between Stirling permutations of size n and chord diagrams with n chords: ζ^{-1} and $\psi \circ \theta \circ \eta^{-1}$. These bijections are highly

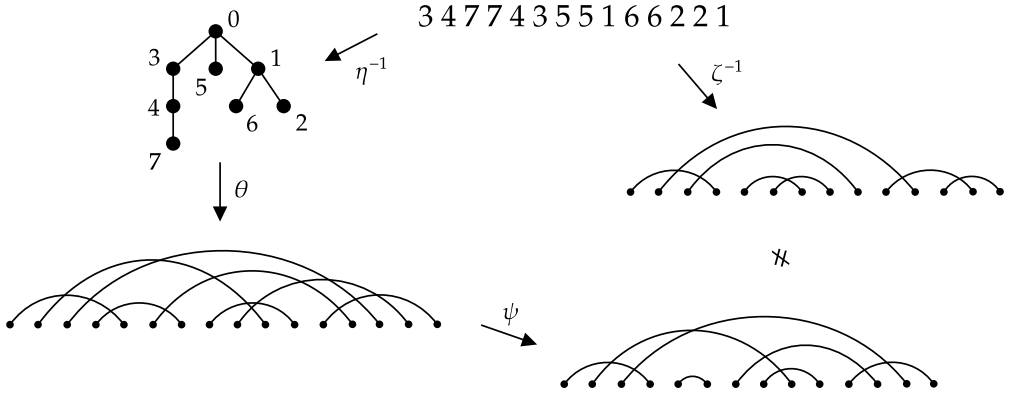


Figure 3.15: A Stirling permutation σ such that $\zeta^{-1}(\sigma) \neq (\psi \circ \theta \circ \eta^{-1})(\sigma)$.

distinct, typically mapping a given Stirling permutation to two different diagrams; e.g. see Figure 3.15. We can think of ζ^{-1} as encoding the *recursive view* of chord diagrams and $\psi \circ \theta \circ \eta^{-1}$ as encoding the *tree structure view* of chord diagrams. The map ψ plays a role in both perspectives, forming a kind of bridge translating 1-terminality from the recursive view to the tree structure view.

3.3.3 Closure under subdiagram avoidance

For a bijection as well-behaved and general as ψ, with its inverse acting on all chord diagrams, it is natural to consider whether ψ underlies enumerative equivalences between proper restricted hereditary subclasses of $\mathcal{T}$ and $\mathcal{D}$. In this section we explain all such equivalences by obtaining a full characterization for when ψ and its inverse are closed under subdiagram avoidance. That is, we determine all sets of diagrams $\mathcal{X}$ such that $\psi(\mathcal{T}(\mathcal{X})) = \mathcal{D}(\mathcal{X})$.

Recall that ψ transforms source-sink groups of a 1-terminal diagram T into sink-source groups of the image $\psi(T)$. For a diagram D, sink-source groups are composed of a source and the left-adjacent interval of sinks, and such these groups partition all points of D excluding the last interval of sinks. We call a sink-source (or source-sink) group *trivial* if it contains no sinks and *nontrivial* otherwise. Note that trivial sink-source groups are trivial source-sink groups and, via the map ψ, nontrivial sink-source groups of D correspond to nontrivial source-sink groups of $\psi^{-1}(D)$ (excluding the terminal chord's group). Furthermore, observe the following basic fact, which we will use throughout the proof of our characterization.

Lemma 3.3.13. *Permutation diagrams are the unique diagrams that contain no nontrivial sink-source groups.*

Proof. Clearly all sink-source groups of a permutation diagram are trivial. In the other direction, the chords attached to a nontrivial sink-source group have  as a subdiagram. □

This suggests that permutation diagrams are "invisible" to the map ψ, since it essentially only effects the points of a nontrivial source-sink group.

We begin our characterization by considering when ψ is closed under subdiagram avoidance. We show that $\psi(\mathcal{T}(X)) \subseteq \mathcal{D}(X)$ for a diagram X if and only if, informally speaking, in a nice sense X is at most two chords away from from being a permutation diagram. For the more difficult "only if" statement, the proof strategy will consist of constructing, for X that are not well-behaved in this sense, diagrams D that contain X and for which $\psi^{-1}(D)$ is X-free. In most cases it suffices to simply take $D = X$, as we now prove.

Define a diagram D to be ***good*** if it is a permutation diagram or there exists $c \in D$ such that 1) c and all later chords (in the standard order) induce a clique and 2) all earlier chords induce a permutation diagram and their sinks lie before the sink of c. We refer to this permutation subdiagram as the ***permutation part*** of D and the rightmost such c as the ***distinguished chord*** of D. In the sequel, we will often abuse notation and use the same variable to refer to a chord in a 1-terminal diagram T (excluding the terminal chord) and its image chord in $\psi(T)$ (or vice versa, equivalently).

Lemma 3.3.14. *For a diagram D, $\psi^{-1}(D)$ is D-free if and only if D is not good.*

Proof. Write $T = \psi^{-1}(D)$. Suppose first that D is good with c the distinguished chord. Since the permutation part of D has no nontrivial sink-source groups, the nontrivial sink-source groups of D can only contain the sources of c and each later chord. By construction, under ψ^{-1} each such chord (except c) gains as new neighbors exactly the missing neighbors of the chord previous to it, while the new terminal chord's neighbors are the neighbors of the last chord d of D plus d itself. It follows that removing c from T leaves D.

For the other direction, suppose D is a subdiagram of T. Then there exists $c \in T$ such that $T - c = D$. If c is the terminal chord then D must be a permutation diagram, so suppose otherwise. The relabeling isomorphism f mapping $T - \{c\}$ to D sends every chord prior to c to itself, implying that each such chord has a trivial source-sink group in T. It follows that all chords lying before c induce a permutation diagram in D. Now suppose c is neither the last chord of D nor a neighbor of the last chord. The terminal

chord of T maps to the last chord d of D under f. Then since the terminal chord is in no nesting, neither is d, so by our supposition the terminal chord crosses every neighbor of d and d itself, implying that it has one more neighbor than d, a contradiction. We infer that c is either the last chord of D or one of its neighbors. Recall that the map ψ preserves nestings, so c is also not in any nesting (since otherwise $T - \{c\}$ is missing a nesting of D); thus, the sink of every $c^{\emptyset} < c$ lies before the sink of c while, by our previous observations, c and all later chords induce a nonnesting subdiagram, in particular necessarily a clique. We conclude that D is good, as desired. □

With this, we obtain the characterization of closure under subdiagram avoidance for ψ. Define a diagram D to be *great* if D is good and either (a) D is permutation, (b) its distinguished chord c has at most 1 neighbor, (c) c is the second-to-last chord, it has at most two neighbors, and the last chord has exactly one neighbor, or (d) $D = \{(1,7),(2,4),(3,6),(5,8)\}$.

Theorem 3.3.15. *For a diagram X, $\psi(\mathcal{T}(X)) \subseteq \mathcal{D}(X)$ if and only if X is great.*

Proof. We begin with the "only if" direction. Suppose X is good but not great and write $X = \{c_1 < \cdots < c_k < \cdots < c_n\}$ with $c_k = c$ the distinguished chord. If $k \leq n-2$, consider the diagram D obtained by nesting a chord as rightmost as possible in X. Then, while the last three chords mutually cross in X, the preimage $\psi^{-1}(D)$ has the second-to-last two chords nesting, implying that one of them must be removed to obtain X. On removal of either chord the terminal chord is reduced to degree one, unlike the last chord of X, so it must also be removed. But c_k still has a trivial sink-source group in the resulting subdiagram $D^{\emptyset}$, unlike in X, so $D^{\emptyset}$ is not X, that is, $\psi^{-1}(D)$ is X-free.

Suppose then that $k = n-1$. If $|N(c_n)| > 2$ then we can repeat the above construction to similarly obtain a diagram D such that $\psi^{-1}(D)$ is X-free. So $|N(c_n)| = 1$, implying that $|N(c_k)| > 3$ since X is not great. Consider the diagram D obtained from X by nesting a chord $c^{\emptyset}$ as rightmost as possible within the source-sink group of c_k. By a similar nesting preservation argument to the previous case either $c^{\emptyset}$ or c_k must be removed from $\psi^{-1}(D)$. Since the terminal chord has degree two while c_n has degree one in D, either the terminal chord, c_n, or c_k must be removed from $\psi^{-1}(D)$. If we do not remove c_k or any earlier chord then we clearly cannot obtain X as a subdiagram, so we may assume that c_k is removed. Furthermore clearly one of $c^{\emptyset}, c_1, \ldots, c_{k-1}$ must be removed from $\psi^{-1}(D)$ since c_k is not part of a nesting in X, unlike $c^{\emptyset}$. But if $c^{\emptyset}$ is not removed then it must map to one of $c_1, \ldots, c_{k-1}$ in X, implying that there is only one sink before the source of $c^{\emptyset}$ in $\psi^{-1}(D)$ and the chord attached to it must be removed. But then $c^{\emptyset}$ maps to c_{k-1}, which is not the bottom chord of a nesting, so we conclude that $\psi^{-1}(D)$ is X-free.

Finally, suppose instead that $k = n$. Since X is not great $|N(c_k)| > 2$. Consider the diagram D obtained from X by nesting a chord c' as rightmost as possible within the source-sink group of c_k and nesting another chord c'' as rightmost as possible within X. If $|N(c_k)| > 3$ then both of the last two chords of $\psi^{-1}(D)$ have fewer left neighbors than c_k in X, so they must be removed. But by nesting preservation the last two chords cannot nest, so we must further remove either c' or c_k. Since neither choice results in X we may assume that $|N(c_k)| = 2$. We now break the argument into two cases: 1) each chord $c_2, \ldots, c_{k-1}$ nests under c_1 and 2) at least one of these chord does not. Suppose first that (1) holds. If either c_k or c'' is removed from $\psi^{-1}(D)$ to get X the terminal chord is reduced to degree at most 1, so we must also remove it. Similarly if both the terminal chord and c'' are removed then we must also remove c'. So if c'' is removed we also remove c' and the terminal chord, but the resulting diagram is not X because c_k remains with more neighbors. We may therefore assume that c'' is not removed, that is, it is in any subdiagram of $\psi^{-1}(D)$ isomorphic to X. By nesting and degree preservation c_k and the terminal chord are either both or neither removed. If the latter holds then c_k maps to c_1 under the subdiagram isomorphism, implying that c_k must be reduced to left degree 0, that is, each of $c_1, \ldots, c_{k-1}$ is removed. But then $|X| = 4$, contradicting the fact that X is not great. So it must be the case that both c_k and and the terminal chord are removed. Then by degree preservation c' and c_1 are not removed, so c' maps to the second neighbor c_ℓ of c_k in X. Thus in order to get X as a subdiagram of $\psi^{-1}(D)$ the chord c_ℓ must have at most one left neighbor and be the top chord of no nestings, implying that $|X| = 4$, again giving a contradiction and thereby resolving (1). Now suppose (2) holds. Considering the same diagram D from before, observe that we may still assume that c'' is not removed and either c_k and the terminal are both removed or neither are removed. If the neighbors of c_k in X cross then it must be the former, since otherwise the nesting (c_k, c'') violates nesting preservation. But again the two neighbors of the last chord c'' nest and neither can be deleted by degree preservation, so we may assume that the neighbors of c_k in X do not cross. Again switching constructions, consider the diagram D obtained from X by nesting a chord c' as rightmost as possible within the source-sink group of c_k and nesting another chord c'' as rightmost as possible within X while crossing the top neighbor of c_k. Observe that while X has one nontrivial sink-source group, the preimage $\psi^{-1}(D)$ has three, so in order to obtain X as a subdiagram we must either delete c', all chords in $\{c_1, \ldots, c_{k-1}\} - N_X(c_k)$, or the terminal chord and one of c'' or the bottom neighbor of c_k in X. Note that if the terminal chord remains we must remove the top neighbor of c_k in X to ensure the neighbors of the last chord do not cross, and if we further remove c' we must also remove c'' to ensure that the last chord has a nontrivial sink-source group. But this leaves the terminal chord with only a single neighbor, so we may assume that either the terminal chord is removed or c' is not removed. It follows that if c' is removed then

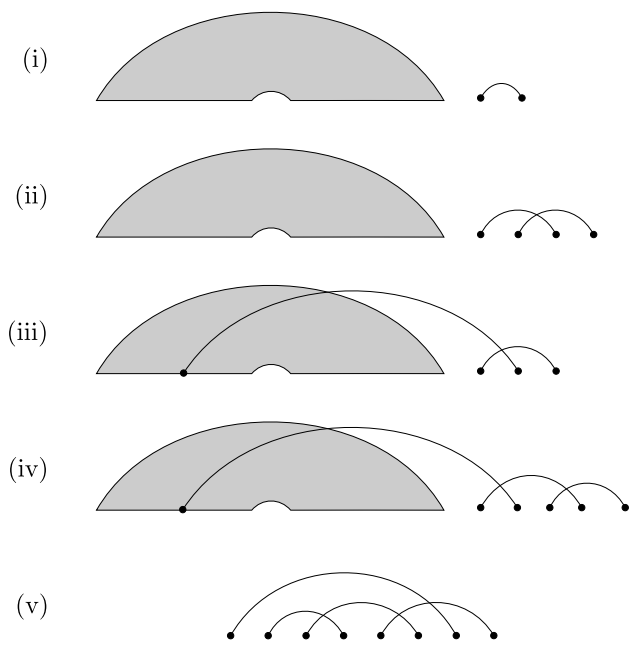


Figure 3.16: The five types of great non-permutation diagrams. The grey section of each diagram indicates the permutation part of the diagram.

$c^{\emptyset\emptyset}$ is also removed by degree preservation, leaving the subdiagram $\{c_1, \ldots, c_k\}$ but with c_k having more neighbors. So we may further assume that $c^{\emptyset}$ is not removed. If the terminal chord and one of $c^{\emptyset\emptyset}$ or the bottom neighbor of c_k are removed, either choice leaves the last chord of degree 1, so it must be removed, implying that the terminal chord, $c^{\emptyset\emptyset}$, and $c^{\emptyset}$ must all be removed. Therefore as before it follows that we may assume that all chords in $\{c_1, \ldots, c_{k\ 1}\} - N_X(c_k)$ are deleted along with the top neighbor of c_k. The remaining chords form a top cycle of size 5, so one can readily check that X must be a top cycle of size 4, but T_4 is not a subdiagram of T_5. Applying Lemma 3.3.14, we conclude that $\psi(\mathcal{T}(X)) \nsubseteq \mathcal{D}(X)$ for all diagrams X such that X is not great, as desired.

Now we turn to the easier "if" direction. Let X be great and, as before, write $X = \{c_1 < \cdots < c_k < \cdots < c_n\}$ with c_k the distinguished chord. It remains to show that $\psi(\mathcal{T}(X)) \subseteq \mathcal{D}(X)$. Consider a diagram D such that X is a subdiagram of D. Since permutation subdiagrams do not contain nontrivial sink-source groups they are preserved by $\psi^{\ 1}$, so we may assume that X is not a permutation diagram. There are five other cases, (i)–(v), visually depicted in Figure 3.16. Since the permutation part is always preserved we need only show the existence of chords in the same configuration as $c_k, \ldots, c_n$. Note that every sink in D is in the source-sink group of some chord in $\psi^{\ 1}(D)$. Furthermore,

Figure 3.17: The two types of near permutation diagrams.

recall that that chord is the rightmost right neighbor of every chord attached to a sink in its source-sink group. It immediately follows that X is a subdiagram of $\psi^{-1}(D)$ in cases (i) and (iii). Furthermore case (ii) also follows since c_n still does not cross or nest $c_1, \ldots, c_{k-1}$ in $\psi^{-1}(D)$. In case (iv), consider the rightmost right neighbor c in $\psi^{-1}(D)$ of the unique left neighbor c_ℓ of c_k in X. Observe that c does not cross or nest $c_1, \ldots, c_{k-1}$. Furthermore c is not the terminal chord, so it must have a right neighbor c' by 1-terminality and any such right neighbor cannot cross c_ℓ. It follows that $\{c_1, \ldots, c_{k-1}, c, c'\}$ is isomorphic to X, as required. Finally, consider case (v). By 1-terminality there exists a nonnesting induced path P from c_3 to the terminal chord in $\psi^{-1}(D)$ and at least one chord in $P - \{c_3\}$ must cross c_1. Letting c be the first such chord, observe that every earlier chord in the path is nested under c_1. Then $P \cup \{c_1, c_2, c_3\}$ contains X, as required. □

We can now complete the full characterization, which follows with substantially less effort. Define a diagram D to be *near permutation* if D consists of a chord nested under a permutation diagram, optionally concatenated with another chord (see Figure 3.17). Observe that near permutation diagrams are great.

Theorem 3.3.16. *For a diagram X, $\psi(\mathcal{T}(X)) = \mathcal{D}(X)$ if and only if X is near permutation.*

Proof. We begin with the "only if" direction. Let X be great but not near permutation and write $X = \{c_1 < \cdots < c_k < \cdots < c_n\}$ with c_k the distinguished chord. Since ψ is bijective it suffices to exhibit a 1-terminal diagram T such that T contains X and $\psi(T)$ is X-free. If X is the diagram from case (v) then we can trivially take $T = X$ since X is 1-terminal. In all other cases X is not necessarily 1-terminal, so take T to be the diagram obtained from X by inserting a chord c whose source lies immediately prior to the sink of c_{k-1} and sink lies immediately after the source of c_k. Then T is clearly 1-terminal and contains X. On the other hand, observe that $c, c_k, \ldots, c_n$ are all isolated in $\psi(T)$ and that $k > 3$ since X is not near permutation. Thus any homomorphism certifying X as a subdiagram of $\psi(T)$ must map c_i to c_i for all $1 \leq i \leq k-1$, but c_{k-1} has left neighbors in X which it loses under ψ, a contradiction.

For the other direction, let X be near permutation and T be a 1-terminal diagram containing X. It suffices to show that $\psi(T)$ contains X. But this follows immediately from

the fact that X contains no nontrivial source-sink groups and, if X is not a permutation diagram, the rightmost chord in T crossing the last chord c in the permutation part of X (which exists and is a right neighbor by 1-terminality) is not the terminal chord and does not cross c in $\psi(T)$. □

Corollary 3.3.17. *For a set $\mathcal{X}$ of diagrams, $\psi(\mathcal{T}(\mathcal{X})) = \mathcal{D}(\mathcal{X})$ if all $X \in \mathcal{X}$ are near permutation.*

This gives an infinite set of distinct pairs of restricted hereditary classes that are enumeratively equivalent and a natural bijection explaining this equivalence.

Corollary 3.3.18. *For all sets $\mathcal{X}$ composed of near permutation diagrams, the map ψ restricts to a bijection between $\mathcal{T}_{n+1}(\mathcal{X})$ and $\mathcal{D}_n(\mathcal{X})$.*

In particular we can immediately count two notable 1-terminal classes.

Corollary 3.3.19. *We have*

$$|\mathcal{T}_{n+1}(\overset{\frown}{\cdot\,\cdot\,\cdot})| = |\mathcal{T}_{n+1}(\overset{\frown}{\cdot\,\cdot\,\cdot})| = C_nC_{n+2} - C_{n+1}^2$$

and

$$|\mathcal{T}_{n+1}(\overset{\frown}{\cdot\,\cdot}\,,\overset{\frown}{\cdot\,\cdot})| = \hat{C}_n,$$

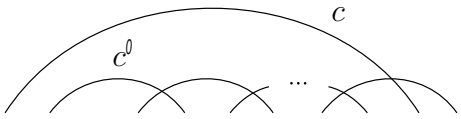
the n^{th} generalized Catalan number.

Proof. These two statements follow immediately from Theorem 3.3.16, the results of Gouyou-Beauchamps [52] and Jelínek [58] enumerating the classes $\mathcal{D}(\overset{\frown}{\cdot\,\cdot\,\cdot})$ and $\mathcal{D}(\overset{\frown}{\cdot\,\cdot\,\cdot})$, respectively, and Theorem 3.1.2 of Bloom and Elizalde [8]. □

3.4 A primer on 1-sym-terminal classes

We finish this chapter with a short note commencing the study of 1-sym-terminal classes. In particular we examine the avoidance of top cycles, bottom cycles, and the diagram $\overset{\frown}{\cdot\,\cdot\,\cdot\,\cdot}$ and its reverse. Further enumeration of other classes of diagrams with the 1-sym-terminal property, including $\mathcal{ST}$ itself, will be discussed in Chapter 5.

We begin with the latter. Observe that for any nesting $(c, c^\emptyset)$ with $c < c^\emptyset$ in a 1-sym-terminal diagram T there must be a nonnesting induced path P beginning with a right neighbor of $c^\emptyset$ and ending in a right neighbor of c by 1-terminality:



The partial top cycle[8] $\{c, c^\emptyset\} \cup P$ contains  as a subdiagram, implying that T does as well. Similarly, 1-rev-terminality gives us such a path involving left neighbors and, thereby,  as a subdiagram of T. Since connected nonnesting diagrams are 1-terminal and their reverses are also connected and nonnesting, this gives the following class equalities.

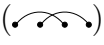
Proposition 3.4.1. *We have*

$$\mathcal{ST}(\text{  }) = \mathcal{ST}(\text{  }) = \mathcal{T}(\text{  }) = \mathcal{C}(\text{  }).$$

Recall that 1-terminal top-cycle-free diagrams are tree diagrams by Theorem 3.3.2. By our above argument any nesting in a 1-sym-terminal diagram is contained in a partial bottom cycle and partial top cycle that combine to form a top cycle of the diagram. Finally, note that all chords in a nonnesting triangle-free diagram have at most a single left neighbor and at most a single right neighbor by the Erdős-Szekeres theorem. Combining these facts, we conclude that nonnesting induced paths are the only 1-sym-terminal top-cycle-free diagrams.

Proposition 3.4.2. *We have* $\mathcal{ST}(T_{>3}) = P_{>1}$.

The class $\mathcal{ST}_{n+2}(B_{>3})$ of 1-sym-terminal bottom-cycle-free diagrams requires somewhat more effort to enumerate but the result is just as nice.

Proposition 3.4.3. *For $n > 0$, there is a bijection between 1-sym-terminal bottom-cycle-free diagrams $\mathcal{ST}_{n+2}(B_{>3})$ of size $n+2$ and noncrossing diagrams $\mathcal{D}_n(\text{  })$ of size n. In particular,*

$$|\mathcal{ST}_{n+2}(B_{>3})| = C_n.$$

Proof. Let T be a 1-sym-terminal bottom-cycle-free diagram of size $n+2$. Note that the left and right neighborhoods of a chord c of T form sets of pairwise nesting chords with no crossings between them. Furthermore, by 1-sym-terminality and our previous observations,

[8] Recall these diagrams from Chapter 2 and several proof trees in the beginning of this chapter and Section 3.1.

only the leftmost (in the standard order) right neighbor and rightmost left neighbor of c may cross chords nested under c. Furthermore, when this occurs, these are the only right and left neighbors, respectively, of the nested chords. Then define $f(T)$ to be the diagram obtained from T as follows:

(1) while preserving the relative order of the sources, move every source of a chord $c \in T$ immediately to the right of the sink of the first chord $c^{\emptyset}$ crossing c, and

(2) delete the root chord and formerly terminal chord.

By our above observations, step (1) removes all crossings, implying that $f(T) \in \mathcal{D}_n(\frown)$. We now exhibit the inverse of the map f, showing that it is a bijection. Let D be a noncrossing diagram of size n and define $g(D)$ to be the diagram obtained from D as follows.

(1') Concatenate a chord to the start and a chord to the end of D.

(2') Let X be a given maximal interval of sources and Y be the maximal interval of sinks adjacent on its left. While preserving the relative order of the sources and sinks, move the first source of X immediately to the left of the first sink of Y and move every other source of X immediately to the left of last sink of Y.

It can be readily verified that this map g produces a 1-sym-terminal bottom-cycle-free diagram and inverts f. □

As Chapter 5 will further support, this begins to get at an offset phenomenon with offset 2 for 1-sym-terminal classes. In that direction, we now define a general map $\chi\colon \mathcal{ST}_{n+2} \to \mathcal{D}_n$ related to ψ and inspired by the above bijection. Consider a 1-sym-terminal diagram T. We refer to the nontrivial source-sink groups of T as the *left point groups* of T and call the nontrivial source-sink groups of T^R viewed as point intervals of T the *right point groups*. Observe the following fact.

Lemma 3.4.4. *For any left point group X and right point group Y of T, X and Y are either (a) disjoint or (b) intersect exactly at an adjacent source and sink.*

Proof. Suppose X and Y are not disjoint. By nontriviality both groups contain at least one source and one sink. Furthermore, by 1-sym-terminality, both groups contain only a single interval of sources and a single interval of sinks, respectively. Then the first sink of X must be the unique sink of Y and the last source of Y must be the unique source of X, implying that (b) holds. □

While this is not quite the "partial partition" property of source-sink groups, we can nevertheless use these point groups to construct χ as a kind of symmetric version of ψ. Define $\chi(T)$ to be the diagram obtained by

(1) moving each source of the left point groups of T to the end of its point group,

(2) moving each sink of the right point groups of T to the start of its point group, and

(3) deleting the root chord and formerly terminal chord.

Observe that, by the above lemma, step (1) does not interfere with the validity of step (2), that is, (2) remains a well-defined transformation. Furthermore, all sources and sinks remain sources and sinks, respectively. Figure 3.18 displays a representative example of T and $\chi(T)$. Step (1) and (2) transform the left and right point groups of T into left and right *reverse point groups* of $\chi(T)$. By construction and Lemma 3.4.4, these are pairwise disjoint and correspond to the nontrivial sink-source groups of a diagram and its reverse.

By construction the map χ also has the nesting preservation property of ψ, but is not injective and correspondingly does not change the number of crossings in as simple a way; e.g. χ sends both of the 1-sym-terminal diagrams of size three,  and  , to the single chord. Nevertheless it does in fact give some information on the counting sequence of $\mathcal{ST}$.

Theorem 3.4.5. *The map χ is a surjection onto $\mathcal{D}_n$ that preserves nestings. In particular, $|\mathcal{ST}_{n+2}| > |\mathcal{D}_n|$.*

Proof. For a diagram $D \in \mathcal{D}_n$, define $\xi(D)$ to be the diagram obtained by

(1') concatenating a chord to the start and and a chord to the end of D

(2') moving each source of the left reverse point groups of D to the start of its reverse point group, and

(3') moving each sink of the right reverse point groups of D to the end of its reverse point group.

Since the reverse point groups are disjoint, steps (1') and (2') are independent. Letting $\mathcal{R} = \xi(D)$, we see that (1'), (2'), and (3') invert (1), (2), and (3) of our construction of ψ, implying that $\psi|_R$ is the inverse of ξ, giving surjectivity. □

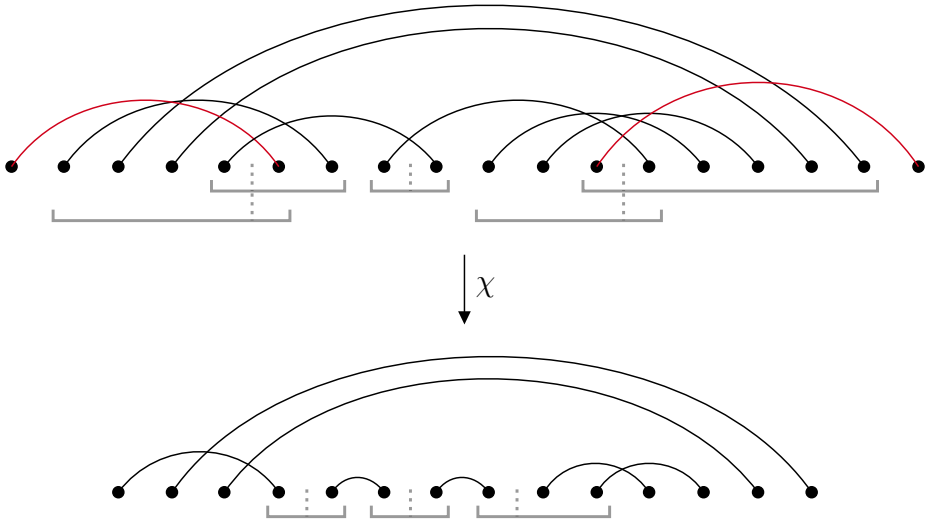


Figure 3.18: An example of a 1-sym-terminal diagram and its image diagram under the map χ. The root and terminal chords are indicated in red and the left and right point groups and their reverses are indicated by the horizontal brackets.

Chapter 4

Dyson-Schwinger generating equations

First appearing under the guise of the Butcher group in numerical analysis and independently introduced by Kreimer [68] in the context of renormalization in perturbative quantum field theory, the Connes-Kreimer Hopf algebra of rooted trees $\mathcal{H}_{CK}$ is the free associative commutative algebra freely generated over a field K of characteristic zero by the set of rooted trees. As implied, the product is given on the basis of forests of rooted trees by concatenation while the coproduct is defined by setting

$$\Delta(t) = \sum_{\substack{C \ \ V(t) \\ C \text{ antichain}}} \left(\prod_{v \in C} t_v \right) \otimes \left(t \setminus \prod_{v \in C} t_v \right)$$

for a rooted tree t, where t_v is the subtree of t rooted at v and parent-child is the cover relation for a tree poset, and extending it as an algebra homomorphism to all of $\mathcal{H}_{CK}$; note that here we take $t \setminus t = 1$. From a pure algebraic perspective, the Connes-Kreimer Hopf algebra is important because it possesses a certain *universal property* unique up to isomorphism among Hopf algebras. Let B_+ be the algebra homomorphism attaching a set of rooted trees as children to a new common root. Note that the B_+ operator is a *Hochschild 1-cocycle*, a linear map L such that

$$(\mathrm{id} \otimes L) \circ \Delta + L \otimes 1 = \Delta \circ L.$$

Then the universal property is the following.

Theorem 4.0.1 (Connes-Kreimer [26, Theorem 2]). *Let A be an associative commutative algebra over K and $L\colon A \to A$ be a linear map. Then there exists a unique algebra homomorphism $\rho_L\colon \mathcal{H}_{CK} \to A$ such that $\rho_L \circ B_+ = L \circ \rho_L$. Furthermore, if A is a bialgebra and L is a Hochschild 1-cocycle then ρ_L is a bialgebra homomorphism, and if A is also a Hopf algebra then ρ_L is a Hopf algebra homomorphism.*

There has been considerable interest in understanding Hopf subalgebras of $\mathcal{H}_{CK}$ (see e.g. [89, 37]). In a series of papers [44, 45, 46], Foissy examined subalgebras of $\mathcal{H}_{CK}$ generated by a family of recursive equations, so-called *combinatorial Dyson-Schwinger equations*, of the form

$$T(x) = xB_+(\phi(T(x))) \tag{4.1}$$

for $\phi(z) \in K[[z]]$ with $\phi(0) = 1$. The unique solution to this equation is a formal power series $T(x)$ whose coefficients lie in $\mathcal{H}_{CK}$. Writing $t_n = [x^n]T(x)$, Foissy characterized when the subalgebra $A = K[t_1, t_2, \ldots]$ of $\mathcal{H}_{CK}$ is Hopf.

Theorem 4.0.2 (Foissy [44]). *A is a Hopf subalgebra if and only if $\phi(z) = (1 + abz)^{1/b}$ for some $a, b \in K$ with $b \neq 0$ or $\phi(z) = e^{az}$.*

We are interested in equations which arise from (4.1) by applying the universal property to the polynomial algebra $K[y]$ and a linear map $L\colon K[y] \to K[y]$. Applying the algebra homomorphism ρ_L guaranteed by Theorem 4.0.1 to both sides of (4.1), we get the bivariate *tree-like equation*

$$G(x, y) = xL(\phi(G(x, y))), \tag{4.2}$$

where $G(x, y) = \rho_L(T(x))$. The maps L and ρ_L act on the coefficients in x of $\phi(G(x, y))$ and $T(x)$ term by term additively; since L sends polynomials to polynomials this equation has an inductively specified solution in $K[y][[x]]$, so it is well-formed. In the physics setting, ρ_L corresponds to the Feynman rules which map each Feynman graph to its associated Feynman integral (for details see e.g. [84]). We will be most interested in Equation (4.2) with ϕ set to generate a Hopf subalgebra of $\mathcal{H}_{CK}$ via Theorem 4.0.2, but will work in the more general setting with ϕ an arbitrary formal power series with constant term 1.

In order to get meaningful combinatorial solutions to (4.2), it is clearly necessary to restrict L to some specific class of linear maps. With that in mind, the universal property points the way towards which classes of maps would be of most interest: Hochschild 1-cocycle operators arising from coalgebra structures on $K[y]$. There are two graded coalgebras on one-variable polynomials classically studied in the literature, namely, the binomial

coalgebra and the divided power coalgebra. For the former, the coproduct is defined by setting

$$\Delta(y^n) = \sum_{k=0}^{n} \binom{n}{k} y^k \otimes y^{n\ \ k}.$$

Combining this with the polynomial algebra on $K[y]$ with the usual product, we in fact get a Hopf algebra. The following lemma describes Hochschild 1-cocycles in the binomial coalgebra.

Lemma 4.0.3. *A map L is a 1-cocycle operator for the binomial coalgebra on $K[y]$ if and only if*

$$L(y^n) = \int_0^y F\left(\frac{d}{dt}\right) t^n dt$$

for some power series $F(z) = \sum_{i>0} f_i z^i$ in $K[[z]]$.

Proof. Writing $c_{m,n} = [y^{n\ \ m}]L(y^n)$ for $m \leq n$, define

$$L_m(y^n) = \begin{cases} c_{m,n} y^{n\ \ m} & \text{if } n > m, \\ 0 & \text{else.} \end{cases}$$

Fix $m \leq n$ and note that L_m is a 1-cocycle by linearity. Then

$$\begin{aligned}((\mathrm{id} \otimes L_m) \circ \Delta + L_m \otimes 1)(y^n) &= (\mathrm{id} \otimes L_m)\left(\sum_{k=0}^{n} \binom{n}{k} y^k \otimes y^{n\ \ k}\right) + L_m(y^n) \otimes 1 \\ &= \sum_{k=0}^{n} c_{m,n\ \ k} \binom{n}{k} y^k \otimes y^{n\ \ m\ \ k} + c_{m,n} y^{n\ \ m} \otimes 1,\end{aligned}$$

while

$$(\Delta \circ L_m)(y^n) = c_{m,n} \Delta(y^{n\ \ m}) = \sum_{i=0}^{n\ \ m} c_{m,n} \binom{n-m}{i} y^i \otimes y^{n\ \ m\ \ i}.$$

Applying the 1-cocycle property and comparing terms, we see that $m > -1$ since otherwise $x^{n+1} \otimes x^{\ \ m\ \ 1}$ appears in the ladder but not the former. Furthermore,

$$c_{m,n} \binom{n-m}{k} = \begin{cases} c_{m,n\ \ k} \binom{n}{k} & \text{if } k \neq n-m, \\ c_{m,n\ \ k} \binom{n}{k} + c_{m,n} & \text{if } k = n-m. \end{cases}$$

It follows that $c_{m,m} = 0$ and, for $m < n$, $c_{m,n} = \frac{n!}{(n\ \ m)!(m+1)!} c_{m,m+1}$. One can then readily check that we get the desired expression for L by setting $f_{m+1} = c_{m,m+1}/(m+1)!$. □

Although we proved it for completeness, this result is well known; e.g. Panzer [84] obtained an equivalent algebraic characterization. For the divided power coalgebra, the coproduct is defined by setting

$$\Delta(y^n) = \sum_{k=0}^{n} y^k \otimes y^{n\ \ k}.$$

This also gives a Hopf algebra on $K[y]$, but the compatible algebra structure instead has the product $y^i \cdot y^j = \binom{i+j}{i} y^{i+j}$; nevertheless it is in fact isomorphic as a Hopf algebra to the binomial Hopf algebra via a scaling of coefficients. With that said, we will always work with the standard product on $K[y]$, meaning that in this case only the first statement of the universal property will apply; we will later see that interesting combinatorics arise regardless. A similar formula holds for Hochschild 1-cocycles in the divided power coalgebra with the integral replaced with a degree raising operator and the derivative with a degree lowering operator δ—its proof can be easily constructed by adapting the proof of Lemma 4.0.3 so we omit it.

Lemma 4.0.4. *A map L is a 1-cocycle operator for the divided power coalgebra on $K[y]$ if and only if*

$$L(y^n) = yF(\delta_y)y^n$$

for some power series $F(z) = \sum_{i>0} f_i z^i$ in $K[[z]]$, where $\delta_y(y^n) = y^{n\ \ 1}$ if $n > 0$ and 0 otherwise.

We will write L_{bin} and L_{div} for 1-cocycles of the binomial and divided power coalgebras, respectively, with the underlying power series F implicitly carried along. Note that for both of these operators $\deg L(y^n)$ 6 $n+1$ and, in particular, this bound is obtained if and only if $f_0 \neq 0$.

4.1 Solving tree-like equations

In this section we will solve tree-like equations arising from a 1-cocycle. The solutions come in the form of weighted generating functions for certain chord diagram classes.

Given a connected diagram C and a sequence of weights $(\phi_k)_{k>0}$ in a field K, we associate a weight ϕ_k with each chord $c \in C$ of valency k (recall Definition 3.2.2 of Section 3.2.1). Then we define the *weight* of C to be

$$\phi_C = \prod_{c2C} \phi_{\mathrm{val}(c)}.$$

This is exactly the same way weights of trees are traditionally defined, at least in the context of tree models and hook lengths [43, 70].

Theorem 4.1.1. *The functional equation*

$$G(x,y) = xL(\phi(G(x,y))) \tag{4.3}$$

is uniquely solved by

$$G(x,y) = \sum_{C \in \mathcal{C}} f_C \phi_C x^{jCj} \frac{L_{bin}(y^{t_1(C)\ 1})}{(t_1(C)-1)!} \quad \text{if} \quad L = L_{bin} \tag{4.4}$$

and

$$G(x,y) = \sum_{C \in \mathcal{C}(T_{>3})} f_C \phi_C x^{jCj} L_{div}(y^{t_1(C)\ 1}) \quad \text{if} \quad L = L_{div}, \tag{4.5}$$

where $\phi(z) = \sum_{k>0} \phi_k z^k$ *for* $\phi_k \in K$ *and*

$$f_C = f_0^{jCj\ \ k} f_{t_2(C)\ \ t_1(C)} f_{t_3(C)\ \ t_2(C)} \cdots f_{t_k(C)\ \ t_k\ \ 1(C)}$$

for a diagram C *with* k *terminal chords and* f_i *the coe cients of the power series* F *specifying* L*.*

In other words, when L is a polynomial 1-cocycle operator Equation (4.3) is solved by a generating function for a certain family of connected weighted chord diagrams where each term is weighted by a monomial in the coefficients defining L determined by the size of the diagram, the number of terminal chords, and the differences between the indices of consecutive terminal chords in the intersection order. The diagrams are counted by their size in the x variable and one less than the index of the first terminal chord in the y variable. While (4.5) can be thought of as ordinary in both variables, (4.4) should be viewed as ordinary in x and exponential in y, although the fact that $L(y^{t_1(C)\ 1})$ appears in the series rather $y^{t_1(C)\ 1}$ makes this less than strictly true. One could account for this by instead regarding both series as ordinary generating functions for a set of polynomials $\{p_C(y)\}$ indexed by a class of weighted chord diagrams. It is notable that the monomial f_C is determined by the gaps between terminal chord indices, not the indices themselves, indicating that it is their relative position that matters, not their absolute position.

Our strategy for proving Theorem 4.1.1 is as follows: 1) show that the decomposition defined in Section 3.2.2 yields suitable weighted versions of the combinatorial identities

in Theorems 3.2.8 and 3.2.9, 2) apply induction to turn these identities into equivalent recurrences for the coefficients in the x variable, and 3) demonstrate that these recurrences are simply an expansion of the corresponding tree-like equations.

We begin with step 1. For this we will need certain statements about how the parameters f_C and ϕ_C behave under the bijection α. Let C be a connected diagram of size n with $t_1(C) = j + 1$, T be its 1-terminal part, and $((C_1, I_1, \ldots, C_k, I_k))$ be the image of C under α. Carry over all other notation from Section 3.2.2 and specifically Definition 3.2.6.

Lemma 4.1.2. *We have*

$$\phi_C = \phi_k \phi_{C_1} \cdots \phi_{C_k}.$$

Proof. Clearly the valency of the terminal chord c_{j+1} is k. Furthermore, for each chord $c \in C_\ell$ the valency of c in C and C_ℓ is equal by construction. Then the identity follows. □

Now let $B_1, \ldots, B_m$ be the components of $C - T$ listed in the intersection order of C; that is, all chords of B_1 come before all chords of B_2, and so on. We require a lemma which says that f_C is determined by $f_D, f_{B_1}, \ldots, f_{B_m}$ and the index of the first terminal chords of $B_1, \ldots, B_m$ in the intersection order of C.

Lemma 4.1.3. *We have*

$$f_C = f_D f_{t_1(B_1)} f_{B_1} \cdots f_{t_1(B_m)} f_{B_m}.$$

In other words, $f_C = f_{C\ B_m} f_{t_1(B_m)} f_{B_m}$.

Proof. By Lemma 1.3.4, terminal chords of B_i are terminal in C and vice versa for all i. Furthermore, by Lemma 1.3.5 T is 1-terminal both as a diagram and as a subdiagram of C. Thus every terminal chord of C corresponds bijectively to a terminal chord in $T, B_1, \ldots, B_k$. Furthermore, clearly the intersection orders on $T, B_1, \ldots, B_k$ agree with the intersection order on C (in the sense that $c \prec c^\emptyset$ in T or B_i if and only if $c \prec c^\emptyset$ in C for $c, c^\emptyset \in T$ or $c, c^\emptyset \in B_i$ for some i) by construction. It follows that the number of terminal chords of C is equal to the sum of the number of terminal chords in T and $B_1, \ldots, B_k$. We also further infer by construction that every difference $t_{i+1}(C) - t_i(C)$ corresponds uniquely to either a difference of consecutive terminal chord indices in some B_i or the difference between the index in C of the last terminal chord of $B_{i\ 1}$ and the index in C of the first terminal chord of B_i, that is, $t_1(B_i)$. The desired equality follows from these two observations. □

This gives the required equality.

Lemma 4.1.4. *We have*

$$f_C = f_{t_1(C_1)\ \ i_1} f_{C_1} \cdots f_{t_1(C_k)\ \ i_k} f_{C_k}.$$

Proof. By Lemma 4.1.3 applied to f_C and $f_{C_1}, \ldots, f_{C_n}$, we may assume that $C - T - \bigcup_\ell A_\ell$ is empty, since the corresponding terms in the monomials can simply be cancelled from both sides. Then $C_\ell = D_\ell \cup A_\ell$. By our observations in the proof of the claim above, the chords of D_ℓ are the first i_ℓ chords of C_ℓ in the intersection order. This implies that

$$f_{C_\ell} = f_{D_\ell [A_\ell} = \begin{cases} f_0^{i_\ell} A_\ell & \text{if } A_\ell \text{ nonempty,} \\ f_0^{i_\ell\ \ 1} & \text{otherwise,} \end{cases}$$

$t_1(A_\ell) = t_1(C_\ell) - i_\ell$ (which is zero if and only if A_ℓ is empty), and

$$f_T = f_0^j = f_0^{i_1 +\ \ + i_k} = f_0^{i_1} \cdots f_0^{i_k}.$$

Then by Lemma 4.1.3

$$\begin{aligned} f_C &= f_T f_{t_1(A_1)} f_{A_1} \cdots f_{t_1(A_k)} f_{A_k} \\ &= f_0^{i_1} \cdots f_0^{i_k} f_{t_1(C_1)\ \ i_1} f_{A_1} \cdots f_{t_1(C_k)\ \ i_k} f_{A_k} \\ &= f_{t_1(C_1)\ \ i_1} f_{C_1} \cdots f_{t_1(C_k)\ \ i_k} f_{C_k}. \end{aligned}$$

□

With this we can give the proof.

Proof of Theorem 4.1.1. Combining the bijection α, the proofs of Theorems 3.2.8 and 3.2.9, and Lemmas 4.1.2 and 4.1.4 we obtain the following identities:

$$\sum_{\substack{C 2 C \\ j C j = n+1 \\ t_1(C) = j+1}} f_C \phi_C = \sum_{k=1}^{n} \phi_k \sum_{\substack{n_1 + \ \ + n_k = n \\ n_\ell > 1}} \sum_{\substack{i_1 + \ \ + i_k = j \\ 1 \leq i_\ell \leq n_\ell}} \binom{j}{i_1, \ldots, i_k} \times \left(\sum_{\substack{C_1 2 C \\ j C_1 j = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k 2 C \\ j C_k j = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k} \phi_{C_k} \right) \tag{4.6}$$

and

$$\sum_{\substack{C 2 C(T_{>3}) \\ j C j = n+1 \\ t_1(C) = j+1}} f_C \phi_C = \sum_{k=0}^{n} \phi_k \sum_{\substack{n_1 + \ \ + n_k = n \\ n_\ell > 1}} \sum_{\substack{i_1 + \ \ + i_k = j \\ 1 \leq i_\ell \leq n_\ell}}$$

$$\times \left(\sum_{\substack{C_1 \mathcal{2C}(T_{>3}) \\ jC_1 j = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \mathcal{2C}(T_{>3}) \\ jC_k j = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k} \phi_{C_k} \right).$$

We require a slightly different but equivalent form of these identities with an extra term added via an additional outermost series on both sides. This essentially matches the left hand side to the innermost series on the right hand side. In particular we instead work with

$$\sum_{j=i\ \ 1}^{n} f_{j+1\ \ i} \sum_{\substack{C \mathcal{2C} \\ jCj = n+1 \\ t_1(C) = j+1}} f_C \phi_C - f_0 \mathbb{1}_{n=0} = \sum_{j=\max\{i\ \ 1,1\}}^{n} f_{j+1\ \ i} \sum_{k=1}^{n} \phi_k \sum_{\substack{n_1 + \ \ + n_k = n \\ n_\ell > 1}} \sum_{\substack{i_1 + \ \ + i_k = j \\ 1 \leq i_\ell \leq n_\ell}} \binom{j}{i_1, \ldots, i_k}$$
$$\times \left(\sum_{\substack{C_1 \mathcal{2C} \\ jC_1 j = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \mathcal{2C} \\ jC_k j = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k} \phi_{C_k} \right)$$

and

$$\sum_{j=i\ \ 1}^{n} f_{j+1\ \ i} \sum_{\substack{C \mathcal{2C}(T_{>3}) \\ jCj = n+1 \\ t_1(C) = j+1}} f_C \phi_C - f_0 \mathbb{1}_{n=0} = \sum_{j=\max\{i\ \ 1,1\}}^{n} f_{j+1\ \ i} \sum_{k=1}^{n} \phi_k \sum_{\substack{n_1 + \ \ + n_k = n \\ n_\ell > 1}} \sum_{\substack{i_1 + \ \ + i_k = j \\ 1 \leq i_\ell \leq n_\ell}}$$
$$\times \left(\sum_{\substack{C_1 \mathcal{2C}(T_{>3}) \\ jC_1 j = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \mathcal{2C}(T_{>3}) \\ jC_k j = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k} \phi_{C_k} \right),$$

where $\mathbb{1}_{n=0} = 1$ if $n = 0$ and 0 otherwise. One can easily check that these are equivalent equalities. Now we translate these identities into polynomial recurrences. Note first that applying L_{bin} to the standard basis of $K[y]$ gives

$$\begin{aligned} L_{bin}(y^n) = \int_0^y F\left(\frac{d}{dt}\right) t^n dt &= \int_0^y \sum_{i>0} f_i \frac{d^i}{dt^i} t^n dt \\ &= \int_0^y \sum_{i=0}^{n} f_i \frac{n!}{(n-i)!} t^{n\ \ i} dt \\ &= n! \sum_{i=1}^{n+1} f_{n+1\ \ i} \frac{y^i}{i!}, \end{aligned} \tag{4.7}$$

while applying L_{div} similarly gives

$$L_{div}(y^n) = yF(d_y)y^n = \sum_{i=1}^{n+1} f_{n+1\ i} y^i. \tag{4.8}$$

Focusing on the binomial 1-cocycle case, define

$$h_n(y) = \sum_{C \in \mathcal{C}_n} f_C \phi_C \frac{L_{bin}(y^{t_1(C)\ 1})}{(t_1(C)-1)!}.$$

We aim to show that

$$h_{n+1}(y) = L_{bin}(1)\mathbb{1}_{n=0} + \sum_{k=1}^{n} \sum_{\substack{n_1 + \cdots + n_k = n \\ n_\ell > 1}} \phi_k L_{bin}(h_{n_1}(y) \cdots h_{n_k}(y)). \tag{4.9}$$

Applying Equation (4.7) to the definition of $h_n(y)$ yields

$$h_n(y) = \sum_{C \in \mathcal{C}_n} f_C \phi_C \sum_{i=1}^{t_1(C)} f_{t_1(C)\ i} \frac{y^i}{i!} = \sum_{i=1}^{n} \left(\sum_{\substack{C \in \mathcal{C} \\ |C| = n \\ t_1(C) > i}} f_{t_1(C)\ i} f_C \phi_C \right) \frac{y^i}{i!},$$

thereby expressing $h_n(y)$ in standard polynomial form. Then

$$\begin{aligned}
&\phi_k L_{bin}(h_{n_1}(y) \cdots h_{n_k}(y)) \\
&\quad = \phi_k L_{bin}\left(\sum_{i_1=0}^{n_1} \left(\sum_{\substack{C_1 \in \mathcal{C} \\ |C_1| = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ i_1} f_{C_1} \phi_{C_1} \right) \frac{y^{i_1}}{i_1!} \cdots \sum_{i_k=0}^{n_k} \left(\sum_{\substack{C_k \in \mathcal{C} \\ |C_k| = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ i_k} f_{C_k} \phi_{C_k} \right) \frac{y^{i_k}}{i_k!} \right) \\
&\quad = \phi_k \sum_{m=0}^{n} L_{bin}(y^m) \sum_{\substack{i_1 + \cdots + i_k = m \\ 0 \le i_\ell \le n_\ell}} \frac{1}{i_1! \cdots i_k!} \left(\sum_{\substack{C_1 \in \mathcal{C} \\ |C_1| = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \in \mathcal{C} \\ |C_k| = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ i_k} f_{C_k} \phi_{C_k} \right) \\
&\quad = \phi_k \sum_{m=0}^{n} \sum_{i=0}^{m+1} f_{m+1\ i} \frac{y^i}{i!} \sum_{\substack{i_1 + \cdots + i_k = m \\ 0 \le i_\ell \le n_\ell}} \frac{m!}{i_1! \cdots i_k!} \\
&\qquad\qquad \times \left(\sum_{\substack{C_1 \in \mathcal{C} \\ |C_1| = n_1 \\ t_1(C_1) > i_1}} f_{t_1(C_1)\ i_1} f_{C_1} \phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \in \mathcal{C} \\ |C_k| = n_k \\ t_1(C_k) > i_k}} f_{t_1(C_k)\ i_k} f_{C_k} \phi_{C_k} \right)
\end{aligned}$$

$$= \sum_{i=0}^{n+1} \frac{y^i}{i!} \sum_{m=\max\{i,\ 1,0\}}^{n} f_{m+1\ \ i}\phi_k \sum_{\substack{i_1+\ +i_k=m \\ 0\leq i_\ell \leq n_\ell}} \binom{m}{i_1,\ldots,i_k} \times \left(\sum_{\substack{C_1 \subseteq C \\ |C_1|=n_1 \\ t_1(C_1)>i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1}\phi_{C_1} \right) \cdots \left(\sum_{\substack{C_k \subseteq C \\ |C_k|=n_k \\ t_1(C_k)>i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k}\phi_{C_k} \right),$$

where the second equality follows from the linearity of L. Consequently, Equation (4.9) follows by applying our earlier combinatorial identity and rearranging and reindexing appropriately. We can perform nearly identical calculations to derive the recurrence

$$h_{n+1}(y) = L_{div}(1)\mathbb{1}_{n=0} + \sum_{k=1}^{n} \sum_{\substack{n_1+\ +n_k=n \\ n_\ell>1}} \phi_k L_{div}(h_{n_1}(y)\cdots h_{n_k}(y))$$

for polynomials $h_n(y) = \sum_{C \subseteq C_n(T_{>3})} f_C \phi_C L_{div}(y^{t_1(C)\ \ 1})$. It remains only to observe that writing $G(x,y) = \sum_{i>1} h_i(y)x^i$ implies that

$$\begin{aligned} G(x,y) &= xL(1) + x\sum_{k>1}\sum_{n>1} \sum_{\substack{n_1+\ +n_k=n \\ n_\ell>1}} \phi_k L(h_{n_1}(y)\cdots h_{n_k}(y))x^n \\ &= x\phi_0 L(1) + xL\left(\sum_{k>1}\sum_{n>1} \sum_{\substack{n_1+\ +n_k=n \\ n_\ell>1}} \phi_k h_{n_1}(y)\cdots h_{n_k}(y)x^n \right) \\ &= xL\left(\sum_{k>0} \phi_k \left(\sum_{i>1} h_i(y)x^i \right)^k \right) \\ &= xL(\phi(G(x,y))) \end{aligned}$$

for L either of the binomial or divided power 1-cocycle, proving that the polynomial recurrences are equivalent to the corresponding tree-like equations. □

4.1.1 A differential equation for the binomial 1-cocycle property

After seeing these results, a natural question immediately arises: why are connected chord diagrams the objects indexed by these generating functions? In this section we present one attempt at an answer to this question, the roots of which go back to the work of Stein

from the 1970s. As previously discussed, the number c_n of connected diagrams of size n satisfies the well known recurrence

$$c_{n+1} = \sum_{k=1}^{n} (2k-1) c_k c_{n+1\ \ k}$$

due to Stein [97]. This corresponds to the differential equation

$$C(x) - x = C(x) \left(2x \frac{d}{dx} - 1 \right) C(x)$$

for the ordinary generating function $C(x)$ of connected diagrams. A bivariate version of this differential equation, known as the *renormalization group equation* in a physics context (see e.g. [110, 111, 37]), turns out to be equivalent to the binomial 1-cocycle property when $\phi(z) = 1/(1-z)$, that is, all the weights are 1. Write $g_i(x) = i![y^i]G(x,y)$.

Theorem 4.1.5. *If G solves the functional equation*

$$G(x,y) = xL(\phi(G(x,y))) \tag{4.10}$$

with $\phi(z) = 1/(1-z)$, then

$$\frac{\partial}{\partial y} G(x,y) = g_1(x) \left(2x \frac{\partial}{\partial x} - 1 \right) G(x,y) \quad \iff \quad L = L_{bin}.$$

Before proving this, we require a lemma which essentially states that if G is the weighted generating function over connected chord diagrams given by Theorem 4.1.1 then it satisfies the differential equation. This lemma was originally proven by Marie and Yeats [75] but we reprove it here for completeness.

Lemma 4.1.6. *For $n > 2$, we have*

$$\sum_{\substack{C \vdash 2C \\ jCj=n \\ t_1(C)>i}} f_{t_1(C)\ \ i} f_C = \sum_{m=1}^{n\ \ i+1} (2(n-m)-1) \sum_{\substack{C_1 \vdash 2C \\ jC_1j=m \\ t_1(C_1)>1}} f_{t_1(C_1)\ \ 1} f_{C_1} \sum_{\substack{C_2 \vdash 2C \\ jC_2j=n\ \ m \\ t_1(C_2)>i\ \ 1}} f_{t_1(C_2)\ \ i+1} f_{C_2}.$$

Proof. Let C be a connected diagram of size n with $t_1(C) > i$. Removing the root chord c_1 of C gives an indecomposable diagram $C^{\emptyset}$; let C_2 be the outermost component of $C^{\emptyset}$, $C_1 = C - C_2$, and i be the endpoint of C_2 immediately left of the leftmost endpoint of $C_1 - \{c_1\}$ in

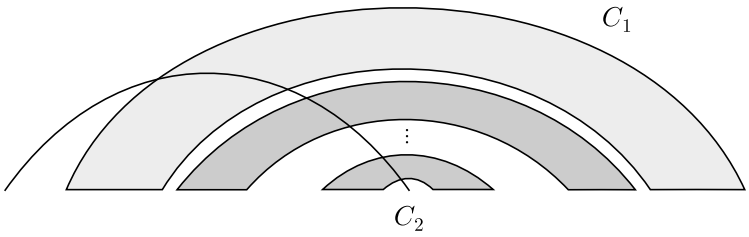


Figure 4.1: The root-share decomposition (C_1, C_2) of C, with the insertion index suppressed.

C. Then (C_1, C_2, i) defines what we previously referred to as the root-share decomposition of C (see Figure 4.1). Clearly we can recover C from (C_1, C_2, i). By construction, C_1 and C_2 are connected. Furthermore, by definition, $t_1(C_2) = t_1(C) - 1 > i - 1$ and, setting $m = |C_1|$, i can take any value in $[1, 2(n-m) - 1]$. Since each terminal chord of C corresponds uniquely to a terminal chord of $C_1 - \{c_1\}$ or C_2, it easily follows that

$$f_C = f_{C_1} f_{t_1(C_1)\ \ 1} f_{C_2}.$$

By all of the above observations, we infer the desired identity. □

Proof of Theorem 4.1.5. We begin with the "only if" direction. Suppose G satisfies the tree-like equation (4.10) and

$$\frac{\partial}{\partial y} G(x,y) = g_1(x) \left(2x \frac{\partial}{\partial x} - 1 \right) G(x,y). \tag{4.11}$$

Write $f_i = \frac{[y] L(y^i)}{i!}$ and $G(x,y) = \sum_{n>1} \left(\sum_{i=1}^{n} g_{i,n} \frac{y^i}{i!} \right) x^n$. We proceed by induction on n to simultaneously show that

$$g_{i,n} = \sum_{\substack{C 2 C \\ jCj=n \\ t_1(C)>i}} f_{t_1(C)\ \ i} f_C \quad \text{and} \quad L(y^n) = n! \sum_{i=1}^{n+1} f_{n+1\ \ i} \frac{y^i}{i!},$$

that is, $L = L_{bin}$. Although proving that G has the form given by Theorem 4.1.1 is not part of the result statement, it seems necessary to generate a straightforward argument.

Where before we began by expanding (4.10) only in x, we now expand in both x and y and then extract coefficients in x to turn the functional equation into the identity

$$\sum_{i=1}^{n+1} g_{i,n+1}\frac{y^i}{i!} = \sum_{j=1}^{n}\left(\sum_{k=1}^{n}\sum_{\substack{n_1+\cdots+n_k=n\\ n_\ell>1}}\sum_{\substack{i_1+\cdots+i_k=j\\ 1\le i_\ell\le m_\ell}}\binom{j}{i_1,\ldots,i_k} g_{i_1,n_1}\cdots g_{i_k,n_k}\right)\frac{L(y^j)}{j!}, \tag{4.12}$$

which holds for all $n > 1$. Applying the induction hypotheses and (4.6), we have

$$\begin{aligned}\sum_{i=1}^{n+1} g_{i,n+1}\frac{y^i}{i!} &= \sum_{j=1}^{n}\left(\sum_{k=1}^{n}\sum_{\substack{n_1+\cdots+n_k=n\\ n_\ell>1}}\sum_{\substack{i_1+\cdots+i_k=j\\ 1\le i_\ell\le m_\ell}}\binom{j}{i_1,\ldots,i_k}\right.\\ &\qquad\left.\times\sum_{\substack{C_1\subset\mathbb{C}\\ |C_1|=n_1\\ t_1(C_1)>i_1}} f_{t_1(C_1)\ i_1 f_{C_1}}\cdots\sum_{\substack{C_k\subset\mathbb{C}\\ |C_k|=n_k\\ t_1(C_k)>i_k}} f_{t_1(C_k)\ i_k f_{C_k}}\right)\sum_{i=1}^{j+1} f_{j+1\ i}\frac{y^i}{i!}\\ &= \sum_{j=1}^{n}\left(\sum_{\substack{C\subset\mathbb{C}\\ |C|=n+1\\ t_1(C)=j+1}} f_C\right)\sum_{i=1}^{j+1} f_{j+1\ i}\frac{y^i}{i!}.\end{aligned}$$

Then upon rearranging we obtain

$$\sum_{i=1}^{n+1} g_{i,n+1}\frac{y^i}{i!} = \sum_{j=1}^{n+1}\left(\sum_{\substack{C\subset\mathbb{C}\\ |C|=n+1\\ t_1(C)>i}} f_{t_1(C)\ i f_C}\right)\frac{y^i}{i!},$$

so extracting coefficients gives the required expression for $g_{i,n+1}$. To derive $L(y^{n+1})$, we begin by noting that differential equation (4.11) expands into the identity

$$g_{i,n} = \sum_{m=1}^{n\ i+1}(2(n-m)-1)g_{1,m}g_{i\ 1,n\ m} \quad \text{for } i > 2. \tag{4.13}$$

Then

$$\sum_{i=1}^{n+2} g_{i,n+2}\frac{y^i}{i!} = \sum_{i=2}^{n+2}\left(\sum_{m=1}^{n\ i+3}(2(n+2-m)-1)g_{1,m}g_{i\ 1,n+2\ m}\right)\frac{y^i}{i!} + g_{1,n+2}y$$

$$= \sum_{i=2}^{n+2} \left(\sum_{m=1}^{n} (2(n+2-m)-1) \sum_{\substack{C_1 2C\ 1 \\ jC_1 j=m \\ t_1(C_1)>1}} f_{t_1(C_1)\ \ i} f_{C_1} \right.$$

$$\left. \times \sum_{\substack{C_2 2C \\ jC_2 j=n+2\ \ m \\ t_1(C_2)>i\ \ 1}} f_{t_1(C_2)\ \ i+1} f_{C_2} \right) \frac{y^i}{i!} + g_{1,n+2} y$$

$$= \sum_{i=2}^{n+2} \left(\sum_{\substack{C2C \\ jCj=n+2 \\ t_1(C)>i}} f_{t_1(C)\ \ i} f_C \right) \frac{y^i}{i!} + g_{1,n+2} y \tag{4.14}$$

by Lemma 4.1.6. On the other hand, applying (4.12) and (4.6) again we get

$$\sum_{i=1}^{n+2} g_{i,n+2} \frac{y^i}{i!} = \sum_{j=1}^{n+1} \left(\sum_{k=1}^{n+1} \sum_{\substack{n_1+ \\ +n_k=n+1 \\ n_\ell>1}} \sum_{\substack{i_1+ \\ +i_k=j \\ 1 6 i_\ell 6 m_\ell}} \binom{j}{i_1, \ldots, i_k} g_{i_1,n_1} \cdots g_{i_k,n_k} \right) \frac{L(y^j)}{j!}$$

$$= \sum_{j=1}^{n+1} \left(\sum_{k=1}^{n+1} \sum_{\substack{n_1+ \\ +n_k=n+1 \\ n_\ell>1}} \sum_{\substack{i_1+ \\ +i_k=j \\ 1 6 i_\ell 6 m_\ell}} \binom{j}{i_1, \ldots, i_k} \right.$$

$$\left. \times \sum_{\substack{C_1 2C \\ jC_1 j=n_1 \\ t_1(C_1)>i_1}} f_{t_1(C_1)\ \ i_1} f_{C_1} \cdots \sum_{\substack{C_k 2C \\ jC_k j=n_k \\ t_1(C_k)>i_k}} f_{t_1(C_k)\ \ i_k} f_{C_k} \right) \frac{L(y^j)}{j!}$$

$$= \sum_{j=1}^{n} \left(\sum_{\substack{C2C \\ jCj=n+2 \\ t_1(C)=j+1}} f_C \right) \sum_{i=1}^{j+1} f_{j+1\ \ i} \frac{y^i}{i!} + \sum_{\substack{C2C \\ jCj=n+2 \\ t_1(C)=n+2}} f_C \frac{f(y^{n+1})}{(n+1)!}$$

$$= \sum_{j=1}^{n+1} \left(\sum_{\substack{C2C \\ jCj=n+2 \\ i 6 t_1(C) 6 n+1}} f_{t_1(C)\ \ i} f_C \right) \frac{y^i}{i!} + \sum_{\substack{C2C \\ jCj=n+2 \\ t_1(C)=n+2}} f_C \frac{f(y^{n+1})}{(n+1)!}. \tag{4.15}$$

Comparing coefficients of (4.15) and (4.14), it follows that

$$[y^i] f(y^{n+1}) = \frac{f_{n+2\ \ i}}{i!}$$

for $i > 2$, as required. We conclude the proof by noting that the "if" direction is directly obtained by combining Equation (4.13), Lemma 4.1.6, and Theorem 4.1.1. □

While the "if" direction of this theorem was proven in [75], the "only if" direction is new. Since in general the weighted count of weighted chord diagrams is only double factorials when the weights are all 1, it is natural for this differential equation to only appear when $\phi(z) = 1/(1-z)$. It is unclear whether analogous differential equations can be obtained in the case of arbitrary weights or as an equivalent condition to the divided power 1-cocycle property.

4.2 Towards generalizing DSEs

There has been considerable work going beyond the simple one-variable combinatorial Dyson-Schwinger equation given in (4.1), starting by generalizing the underlying Hopf algebra. Given a combinatorial class $\mathcal{J}$ of so-called *decorations*, we form a decorated version $\mathcal{H}_{CK}^{\mathcal{J}}$ of the Connes-Kreimer Hopf algebra by associating each vertex in a rooted tree with a decoration from $\mathcal{J}$ and defining $\mathcal{H}_{CK}^{\mathcal{J}}$ to be the free associative commutative algebra freely generated over a field K of characteristic zero by the set of decorated rooted trees. Algebraically, the size function on the class $\mathcal{J}$ makes it a graded set and correspondingly $\mathcal{H}_{CK}^{\mathcal{J}}$ a graded Hopf algebra. The coproduct $\Delta_{\mathcal{J}}$ is constructed similarly to the undecorated version. The decorated Connes-Kreimer Hopf algebra also satisfies a universal property. Let B_+^j be the decorated version of B_+ attaching a set of decorated rooted trees as children to a new common root decorated by a fixed $j \in \mathcal{J}$.

Theorem 4.2.1 (van der Laan and Moerdijk [103])**.** *Let A be an associative commutative algebra over K and, for all $j \in \mathcal{J}$, let $L_j \colon A \to A$ be a linear map. Then there exists a unique algebra homomorphism $\rho \colon \mathcal{H}_{CK}^{\mathcal{J}} \to A$ such that for all $j \in \mathcal{J}$, $\rho \circ B_+^j = L_j \circ \rho$. Furthermore, if A is a bialgebra and, for all $j \in \mathcal{J}$, L_j is a Hochschild 1-cocycle then ρ is a bialgebra homomorphism, and if A is also a Hopf algebra then ρ is a Hopf algebra homomorphism.*

Generalizing his prior work, in [45, 46, 47, 48] Foissy investigated systems of combinatorial Dyson-Schwinger equations of the form

$$T_1(x) = \sum_{j \in \mathcal{J}_1} x^{jjj} B_+^{j,1}(\phi_{j,1}(T_1(x), \ldots, T_m(x)))$$

$$
\begin{aligned}
T_2(x) &= \sum_{j \in J_2} x^{jj} B_+^{j,2}(\phi_{j,2}(T_1(x), \dots, T_m(x))) \\
&\vdots \\
T_m(x) &= \sum_{j \in J_m} x^{jj} B_+^{j,m}(\phi_{j,m}(T_1(x), \dots, T_m(x)))
\end{aligned}
\tag{4.16}
$$

for a partition $\mathcal{J}_1 \sqcup \mathcal{J}_2 \sqcup \cdots \sqcup \mathcal{J}_m$ of $\mathcal{J}$ and, for all $j \in \mathcal{J}$ and $1 \leq i \leq m$, $\phi_{j,i}(z) \in K[[z_1, \dots, z_m]]$ with $\phi_{j,i}(0) = 1$. The unique solution to this system is a tuple of formal power series $(T_1(x), \dots, T_m(x))$ whose coefficients lie in $\mathcal{H}_{CK}^{\mathcal{J}}$. Writing $t_{i,n} = [x^n]T_i(x)$, as previously Foissy characterized when the subalgebra $A = K[t_{i,n}, 1 \leq i \leq m, 1 \leq n]$ of $\mathcal{H}_{CK}^{\mathcal{J}}$ is Hopf. The full solution is much more complex than in the undecorated single equation case; see [45, 46, 47] for the details. If we reduce to a single equation while keeping the decorations than it comes in a form quite close to Theorem 4.0.2, with an extra linear case based on a divisibility property. We note that Rotheray [89] further generalized this work in a particularly physically relevant direction.

As before, we consider equations which arise from (4.16) by applying the universal property to the polynomial algebra $K[y]$ and linear maps $L_{j,i} \colon K[y] \to K[y]$ for all $j \in \mathcal{J}$ and $1 \leq i \leq m$. Applying the algebra homomorphism ρ from Theorem 4.2.1 to both sides of each equation in (4.16), we get the bivariate system of tree-like equations

$$
\begin{aligned}
G_1(x, y) &= \sum_{j \in J} x^{jj} L_{j,1}(\phi_{j,1}(G_1(x, y), \dots, G_m(x, y))) \\
&\vdots \\
G_m(x, y) &= \sum_{j \in J} x^{jj} L_{j,m}(\phi_{j,m}(G_1(x, y), \dots, G_m(x, y))),
\end{aligned}
\tag{4.17}
$$

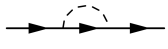
where $G_i(x, y) = \rho(T_i(x))$. It is of most interest to work with this system with the formal power series $\phi_{j,i}$ set to generate a Hopf subalgebra of $\mathcal{H}_{CK}^{\mathcal{J}}$ via Foissy's work, but we remain in the general setting. As previously, for both algebraic and quantum field theoretic reasons we study systems of tree-like equations with the maps $L_{j,i}$ being Hochschild 1-cocycles or closely related operators.

4.2.1 Analytic Dyson-Schwinger equations, briefly

We now briefly expand on the relationship between systems of tree-like equations and quantum field theory, a key motivation and application for much of this work. Our explanation will stay at a high level, employing less than full physical or analytic rigor. See

[15, 110, 84, 111] for more details. The combinatorial Dyson-Schwinger equation with $\phi(z) = \frac{1}{1-z}$,

$$T(x) = xB_+\left(\frac{1}{1-T(x)}\right),$$

can also be interpreted as specifying all possible ways of inserting the primitive[1] Feynman graph  into itself.[2] As mentioned previously, the homomorphism ρ corresponds to the Feynman rules which map each Feynman graph to its associated Feynman integral. The tree-like equation (4.2) with $\phi(z) = \frac{1}{1-z}$ then corresponds to the *analytic Dyson-Schwinger equation* expressed in massless Yukawa theory as

$$G(x,y) = 1 - \left(\frac{x}{q^2}\int d^4k \frac{kq}{k^2(k+q)^2 G(x,\log(k^2/\mu^2))} - \text{same integrand}\Big|_{q^2=\mu^2}\right),$$

where $y = \log(q^2/\mu^2)$, x is the so-called coupling constant, q is the momentum going through a Feynman graph, renormalization of the Feynman integral is taking place by subtraction at a fixed reference scale μ, and $G(x,y)$ is the Green function for the fermion propagator.[3] The Feynman rules transformed the formal power series $T(x)$ into the Green function $G(x,y)$. It should be emphasized that this is an example of an analytic Dyson-Schwinger equation in a particularly heavily restricted quantum field theory. This equation was first solved using sophisticated analytic methods by Broadhurst and Kreimer [15]. In [110] this equation was transformed into a simpler pseudo-differential form with the analytic contributions packed into a single term:

$$G(x,y) = 1 - xG\left(x, \frac{\partial}{\partial(-\nu)}\right)^{1}(e^{y\nu} - 1)F(\nu)\Big|_{\nu=0},$$

where $F(\nu)$ can now be taken to be the Laurent series expansion $\sum_{i>0} f_i \nu^i$ [1] of the Feynman integral of the primitive graph. This is simply an alternative way of expressing this tree-like equation; in particular,

$$\frac{\partial^n}{\partial(-\nu)^n}(e^{y\nu} - 1)F(\nu)\Big|_{\nu=0} = \int_0^y F\left(\frac{d}{dt}\right)t^n dt$$

[1] As in, primitive in the associated Hopf algebra of Feynman graphs.

[2] Note that for a Feynman graph we view each edge as two half-edges of a specific type and when inserting a diagram into itself half-edge types must match, so in this case there is only one possible insertion place.

[3] The directed half-edges in the primitive graph are fermion propagators.

is the action of a 1-cocycle on the standard basis polynomial y^n (recall Lemma 4.0.3). Then extending to tree-like systems (4.17) corresponds to generalizing to a more physically realistic setting involved a coupled system of analytic Dyson-Schwinger equations with multiple particle types (and, thereby, multiple Green functions, one for each equation), multiple primitive Feynman graphs, and multiple distinguishable places (that is, edges[4]) for the primitive diagrams to insert into themselves. The sizes of the decorations j in the class $\mathcal{J}$ correspond to the cycle ranks, that is, dimension of the cycle space, of the primitive graphs. Since the number of insertion places in a primitive graph of cycle rank k is $sk-1$ [110], incorporating different insertion places into the counting leads to the following system:

$$\begin{aligned}
G_1(x,y) = 1 - \sum_{k>1} x^k \Bigg(\prod_{i=1}^{s_1 k - 1} G_1\left(x, \frac{\partial}{\partial(-\nu_{1,i})}\right) \prod_{\ell \neq 1} \prod_{j=1}^{s_\ell k} G_\ell\left(x, \frac{\partial}{\partial(-\nu_{\ell,j})}\right) \Bigg) \\
\times \left(e^{L\sum_{i=1}^{s_1 k - 1} \nu_i} - 1\right) F_{1,k}(\nu_1, \ldots, \nu_{s_1 k - 1}) \Big|_{\nu_1 = \cdots = \nu_{s_1 k - 1} = 0} \\
\vdots \\
G_m(x,y) = 1 - \sum_{k>1} x^k \Bigg(\prod_{i=1}^{s_m k - 1} G_m\left(x, \frac{\partial}{\partial(-\nu_{m,i})}\right) \prod_{\ell \neq m} \prod_{j=1}^{s_\ell k} G_\ell\left(x, \frac{\partial}{\partial(-\nu_{\ell,j})}\right) \Bigg) \\
\times \left(e^{L\sum_{i=1}^{s_m k - 1} \nu_i} - 1\right) F_{m,k}(\nu_1, \ldots, \nu_{s_m k - 1}) \Big|_{\nu_1 = \cdots = \nu_{s_m k - 1} = 0} .
\end{aligned} \tag{4.18}$$

4.2.2 Solving generalized tree-like systems

An ambitious goal is to uniquely solve the system (4.17) of tree-like equations via bivariate generating functions $G_i(x,y)$ for some class of weighted, decorated combinatorial objects. Here we report several conjectures towards obtaining such solutions.

Computer calculations suggest that there is a simple transformation turning a solution for the tree-like equation (4.2) into a solution to the version where the map L physically allows for multiple distinguishable insertion places. For a positive integer n, write $n!^{(m)} = n(n-m)(n-2m)\cdots 1$ for the m-multifactorial of n.

[4]Inserting at vertices is also possible in more physically realistic quantum field theories, but the framework given here is restricted to edge insertions.

Conjecture 4.2.2. *Fix an integer $s > 2$ and let G be a solution to*

$$G(x,y) = xL_{bin}(\phi(G(x,y))),$$

where

$$\phi(z) = \frac{1}{(1-z)^{s^0-1}}$$

for $s^0 = 2s-1$. Then $\hat{G}$ obtained by applying the substitution

$$f_i \to \hat{f}_i := \sum_{j_1+\cdots+j_{s-1}=i} \frac{\prod_{l=1}^{s-1}(s(j_l-1)+1)!^{(s)}}{(s^0 i-1)!^{(s^0)}} f_{j_1\cdots j_{s-1}} \quad \forall i > 0 \tag{4.19}$$

to G uniquely solves

$$\hat{G}(x,y) = xL_s\left(\frac{1}{1-\hat{G}(x,y)}\right), \tag{4.20}$$

where

$$L_s(y^n) = \left(\prod_{i=1}^{s-1}\frac{\partial^n}{\partial(-\nu_i)^n}\right)(e^{y\sum_{i=1}^{s-1}\nu_i}-1)F(\nu_1,\ldots,\nu_{s-1})\Big|_{\nu_1=\cdots=\nu_{s-1}=0}.$$

While we have not proven this conjecture, combined with (4.1.1) the multifactorial-based substitution points towards the unique solution to (4.20) taking the form of a generating function for a class of combinatorial objects closely related to chord diagrams.

Turning to the more general context of systems of the form (4.18), for quantum field theoretic reasons it is expected that solutions to physically-realistic analytic Dyson-Schwinger equations generally satisfy a renormalization group equation as studied in Section 4.1.1. Obtaining such renormalization group equations has the potential to both help discover generating function solutions to tree-like systems and, if necessary, assist in proving that they are indeed solutions, as featured in [75, 54]. Based upon Theorem 4.1 of [110] and extensive computer calculations of the initial terms of solutions to these type of equations, we have the following conjecture.

Conjecture 4.2.3. *Every solution $(G_1,\ldots,G_m)$ to (4.18) with $s_i = s$ for a fixed integer $s > 2$ satisfies the system*

$$\left(\frac{\partial}{\partial y} + \beta_1(x)\frac{\partial}{\partial x} - \gamma_1(x)\right)G_1(x,y) = 0$$

$$\vdots$$

$$\left(\frac{\partial}{\partial y} + \beta_m(x)\frac{\partial}{\partial x} - \gamma_m(x)\right) G_m(x,y) = 0$$

of renormalization group equations with $\gamma_{i,n}(x) = [y^n]G(x,y)$*,* $\gamma_i(x) = \gamma_{i,1}(x)$*, and* $\beta_i(x) = sx\sum_{i=1}^{m}\gamma_{i,1}(x)$*.*

Chapter 5

Conjectures, speculations, and conclusions

We developed several enumeration devices, most notably an extension of the Combinatorial Exploration framework [1] to chord diagrams, that have the potential to enumerate or at least give structural information about many more classes of chord diagrams than we considered in this thesis. Towards that goal, we computed the first 8 terms of the counting sequences of approximately $10,000$ restricted hereditary diagram classes obtained by applying combinations of the following properties:

- connectivity, 1-terminality, 1-sym-terminality
- cycle-free, top-cycle-free, bottom-cycle-free, bipartite, chordal
- avoiding some number of permutation diagrams of size 3,
- avoiding one or two permutation diagrams of size 4
- avoiding a cycle of size 4 or 5

This reduces to approximately $4,500$ classes after partially accounting for symmetries and non-minimal forbidden diagram combinations. Around $1,100$ of the sequences match sequences found on the OEIS [93], indicating that they are known and likely related to objects that have been studied in the literature or, at least, algebraic or analytic information can be acquired on the sequence. Some notable sequences include the following (excluding the first couple terms):

- 199 occurrences of the all 1s sequence: $1, 1, 1, 1, 1, 1, \ldots$ [93, A000012]
- 163 occurrences of the Catalan numbers: $2, 5, 14, 42, 132, 429, \ldots$ [93, A000108]
- 90 occurrences of a bisection of the Fibonacci numbers: $2, 5, 13, 34, 89, 233, \ldots$ [93, A001519]
- 41 occurrences of the powers of 2: $2, 4, 8, 16, 32, 64, \ldots$ [93, A000079]
- 27 occurrences of the Motzkin numbers: $2, 4, 9, 21, 51, 127, \ldots$ [93, A001006]
- 22 occurrences of the Fibonacci numbers: $2, 3, 5, 8, 13, 21, \ldots$ [93, A000045]
- 20 occurrences of the number of $(1243, 2134)$-avoiding permutations: $2, 6, 22, 87, 354, 1459, \ldots$ [93, A164651]
- 20 occurrences of the number of $(2341, 3421)$-avoiding permutations: $2, 6, 22, 89, 382, 1711, \ldots$ [93, A165545]
- 20 occurrences of the number of $(4123, 3412)$-avoiding permutations: $2, 6, 22, 89, 381, 1696, \ldots$ [93, A165544]
- 18 occurrences of $\frac{1}{2n+1}\binom{3n}{n}$: $3, 12, 55, 273, 1428, 7752, \ldots$ [93, A001764]
- 18 occurrences of the little Schröder numbers: $3, 11, 45, 197, 903, 4279, \ldots$ [93, A001003]

In Table 5.1 we give a sampling of some of the most intriguing conjectures that fall out of this, incorporating the conjectured "proper" size indexing of the class, i.e. conjectured offset phenomenon applicable.

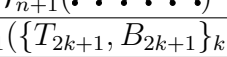
$\mathcal{C}_{n+1}(T_{>4}, B_{>4})$	C_n^2 [93, A001246]
$\mathcal{T}_{n+1}($  $)$	[93, A117106]
$\mathcal{T}_{n+1}(\{T_{2k+1}, B_{2k+1}\}_{k>1})$	[93, A001181]
$\mathcal{ST}_{n+2}$	[93, A001515, A144301]
$\mathcal{ST}_{n+2}(T_3)$	[93, A108304]
$\mathcal{ST}_{n+1}(T_{>4}, B_{>4})$	[93, A005802]

Table 5.1: A sampling of diagram sets and their conjectured counting sequences.

To prove these conjectures it would obviously be prudent to expand the pool of combinatorial strategies for our version of Combinatorial Exploration for chord diagrams and,

most importantly, implement it on a computer. This has the potential to resolve many or all of these conjectures. It would also be most interesting to get at bijections, or even a single universal map $C_{n+1} \to \mathcal{D}_n$, explaining the offset phenomenon for connected and 1-terminal classes. This may extend to explaining the offset 2 phenomenon of 1-sym-terminal classes, or a different map may be needed, such as the one presented in Section 3.4.

Finally, it would be of interest to resolve the conjectures of Section 4.2.2 and obtain generating functions for chord diagrams or other combinatorial objects that solve generalized Dyson-Schwinger equations, such as those described in Section 4.2.

References

[1] M. H. Albert, C. Bean, A. Claesson, E. Nadeau, J. Pantone, and H. Ulfarsson. Combinatorial Exploration: An algorithmic framework for enumeration.

[2] E. Barcucci, A. D. Lungo, E. Pergola, and R. Pinzani. ECO: a methodology for the enumeration of combinatorial objects. *J. Di . Eq. Appl.*, 5:435–490, 1999.

[3] J. S. Beissinger. The enumeration of irreducible combinatorial objects. *J. Combin. Theory Ser. A*, 38(2):143–169, 1985.

[4] F. Bergeron, P. Flajolet, and B. Salvy. Varieties of increasing trees. In J. C. Raoult, editor, *CAAP '92*, pages 24–48, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg.

[5] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial Species and Tree-like Structures, Encyclopedia of Mathematics and its Applications*, volume 67. Cambridge University Press, 1998.

[6] O. Bernardi and N. Bonichon. Intervals in Catalan lattices and realizers of triangulations. *J. Combin. Theory Ser. A*, 116(1):55–75, 2009.

[7] G. Birkhoff. *Lattice Theory*, volume 25 of *Amer. Math. Soc. Colloq. Pub*. American Mathematical Society, 1967.

[8] J. Bloom and S. Elizalde. Pattern avoidance in matchings and partitions. *Electron. J. Combin*, 20(2), 2013.

[9] J. Bloom and D. Saracino. Pattern avoidance for set partitions á la Klazar. *Discrete Math and Theor. Comput. Sci*, 18(2), 2016.

[10] B. Bollobás and O. Riordan. Linearized chord diagrams and an upper bound for Vassiliev invariants. *J. Knot Theory Rami cations*, 9(7):847–853, 2000.

[11] M. Bona. Exact enumeration of 1342-avoiding permutations: A close link with labelled trees and planar maps. *J. Combin. Theory Ser. A*, 175:55–67, 1997.

[12] M. Bóna. Real zeros and normal distribution for statistics on Stirling permutations defined by Gessel and Stanley. *SIAM J. Discrete Math*, 23(1):401–406, 2009.

[13] A. Bouchet. Circle graph obstructions. *J. Combin. Theory Ser. B*, 60(1):107–144, 1994.

[14] M. Bouvel, V. Guerrini, A. Rechnitzer, and S. Rinaldi. Semi-Baxter and strong-Baxter: two relatives of the Baxter sequence. *SIAM J. Discrete Math*, 32(4):2795–2819, 2018.

[15] D. J. Broadhurst and D. Kreimer. Exact solutions of Dyson-Schwinger equations for iterated one-loop integrals and propagator-coupling duality. *Nuclear Physics B*, 600:403–422, 2001.

[16] W. G. Brown. Enumeration of triangulations of the disk. *Proc. Lond. Math. Soc*, 3(14):746–768, 1964.

[17] S. Burrill. Crossings and nestings in four combinatorial families. Master's thesis, Simon Fraser University, 2007.

[18] A. Burstein. Restricted Dumont permutations. *Ann. Comb*, 9(3):169–280, 2005.

[19] Y. Cai and C. Yan. Counting with Borel's triangle. *Discrete Math.*, 342(2):529–539, 2019.

[20] L. R. Campbell, S. Dahlberg, R. Dorward, J. Gerhard, T. Grubb, C. Purcell, and B. E. Sagan. Restricted growth function patterns and statistics. *Adv. Appl. Math*, 100:1–42, 2018.

[21] M. Cervetti and L. Ferrari. *Pattern avoidance in the matching pattern poset*, 2020.

[22] F. Chapoton. Sur le nombre d'intervalles dans les treillis de Tamari. *Sem. Lothar. Combin*, 55, 2006.

[23] W. Y. C. Chen, E. Y. P. Deng, R. R. X. Du, R. P. Stanley, and C. H. Yan. Crossings and nestings of matchings and partitions. *Trans. Amer. Math. Soc*, 359(4):1555–1575, 2007.

[24] W. Y. C. Chen, T. Mansour, and S. H. F. Yan. Matchings avoiding partial patterns. *Electron. J. Combin*, 13, 2006.

[25] F. R. K. Chung, R. L. Graham, E. Hoggatt, and M. Kleiman. The number of Baxter permutations. *J. Combin. Theory Ser. A*, 24:382–394, 1978.

[26] A. Connes and D. Kreimer. Hopf algebras, renormalization and noncommutative geometry. *Comm. Math. Phys*, 199:203–242, 1998.

[27] R. Cori, S. Dulucq, and G. Viennot. Shuffle of parenthesis systems and Baxter permutations. *J. Combin. Theory Ser. A*, 43(1):1–22, 1986.

[28] J. Courtiel and K. Yeats. Terminal chords in connected chord diagrams. *Ann. Inst. Henri Poincare Comb. Phys. Interact*, 4(4):417–452, 2017.

[29] J. Courtiel and K. Yeats. *Next-to^k leading log expansions by chord diagrams*, 2019.

[30] J. Courtiel, K. Yeats, and N. Zeilberger. Connected chord diagrams and bridgeless maps. *Electron. J. Combin*, 26(4), 2019.

[31] S. Dahlberg, R. Dorward, J. Gerhard, T. Grubb, C. Purcell, L. Reppuhn, and B. E. Sagan. Set partition patterns and statistics. *Discrete Math*, 339(1):1–16, 2016.

[32] J. Davies and R. McCarty. Circle graphs are quadratically χ-bounded. *Bull. Lond. Math. Soc*, 53(3):673–679, June 2021.

[33] M. de Sainte-Catherine. *Couplages et Pfa ens en combinatoire, physique et informatique*. PhD thesis, University of Bordeaux, 1983.

[34] M. de Sainte-Catherine and G. Viennot. Enumeration of certain Young tableaux with bounded height. *Lecture Notes in Math*, 1234:58–67, 1986.

[35] C. Defant. Catalan intervals and uniquely sorted permutations. *J. Combin. Theory Ser. A*, 174, 2020.

[36] C. Defant, M. Engen, and J. A. Miller. Stack-sorting, set partitions, and Lassalle's sequence. *J. Combin. Theory Ser. A*, 175, 2020.

[37] W. Dugan. Sequences of trees and higher-order renormalization group equations. Master's thesis, University of Waterloo, 2019.

[38] S. Dulucq and J.-G. Penaud. Cordes, arbres et permutations. *Discrete Math*, 117(1):89–105, 1993.

[39] D. Dumont. Interpretations combinatoires des nombres de Genocchi. *Duke J. Math*, 41:305–218, 1974.

[40] P. Erdos and G. Szekeres. A combinatorial problem in geometry. *Comp. Math*, 2:463–470.

[41] W. Fang. Planar triangulations, bridgeless planar maps and Tamari intervals. *European J. Combin*, 70:75–91, 2018.

[42] P. Flajolet and M. Noy. Analytic combinatorics of chord diagrams. In *Formal Power Series and Algebraic Combinatorics*, pages 191–201. Springer, 2000.

[43] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge Univ. Press, Cambridge, 2009.

[44] L. Foissy. Faá di Bruno subalgebras of the [hopf. *Adv. Math*, (1):136–162.

[45] L. Foissy. Classification of systems of Dyson-Schwinger equations in the Hopf algebra of decorated rooted trees. *Adv. Math*, 224(5):2094–2150, 2010.

[46] L. Foissy. General Dyson-Schwinger equations and systems. *Comm. Math. Phys*, 327(1):151–179, 2014.

[47] L. Foissy. Pre-lie algebras and systems of dyson-schwinger equations. In *Dyson-Schwinger Equations and Faa di Bruno Hopf Algebas in Physics and Combinatorics*, pages 9–89. European Mathematical Society, 2016.

[48] L. Foissy. Multigraded Dyson–Schwinger systems. *J. Math. Phys*, 61(5):051703, 2020.

[49] E. Fusy. New bijective links on planar maps via orientations. *European J. Combin*, 31(1):145–160, 2010.

[50] I. Gessel and R. P. Stanley. Stirling polynomials. *J. Combin. Theory Ser. A*, 24:24–33, 1978.

[51] I. M. Gessel. Symmetric functions and P-recursiveness. *J. Combin. Theory Ser. A*, 53:257–285, 1990.

[52] D. Gouyou-Beauchamps. Standard young tableaux of height 4 and 5. *European J. Combin*, 10:69–82, 1989.

[53] J. M. Hammersley. A few seedlings of research. In *Proc. Sixth Berkeley Sym. Math. Stat. Prob.*, volume 1, pages 345–394, Berkley/Los Angeles, 1972. University of California Press.

[54] M. Hihn and K. Yeats. Generalized chord diagram expansions of Dyson-Schwinger equations. *Ann. Inst. Henri Poincare Comb. Phys. Interact*, 6(4):573–605, 2019.

[55] I. Hofacker, P. Schuster, and P. F. Stadler. Combinatorics of RNA secondary structures. *Discrete Appl. Math*, 88(1–3):207–237, 1998.

[56] S. Janson. Plane recursive trees, Stirling permutations and an urn model. *Fifth Colloquium on Mathematics and Computer Science*, pages 541–548, 2008.

[57] S. Janson, M. Kuba, and A. Panholzer. Generalized Stirling permutations, families of increasing trees and urn models. *J. Combin. Theory Ser. A*, 118(1):94–114, 2011.

[58] V. Jelínek. Dyck paths and pattern-avoiding matchings. *European J. Combin*, 28:202–213, 2005.

[59] V. Jelínek, T. Mansour, and M. Shattuck. On multiple pattern avoiding set partitions. *Adv. Appl. Math*, 50(2):292–326, 2013.

[60] A. Joyal. Calcul intégral combinatoire et homologie du groupe symétrique. *Comptes Rendus Math. Acad. Sci.*, 7:337–342, 1985.

[61] A. Joyal. Règle des signes en algèbre combinatoire. *Comptes Rendus Math. Acad. Sci.*, 7:285–290, 1985.

[62] S. Kitaev. *Patterns in Permutations and Words*. Springer-Verlag, Berlin, 2011.

[63] S. Kitaev and T. Mansour. A survey on certain pattern problems. 2007.

[64] M. Klazar. On *abab*-free and *abba*-free set partitions. *European J. Combin*, 17:53–68, 1996.

[65] M. Klazar. Counting pattern-free set partitions I: A generalization of Stirling numbers of the second kind. *European J. Combin*, 21:367–378, 2000.

[66] D. J. Kleitman. Proportions of irreducible diagrams. *Stud. Appl. Math*, 44(3):297–299, 1970.

[67] D. E. Knuth. *The Art of Computer Programming, vol. 1, Fundamental Algorithms*. Addison-Wesley, 1973.

[68] D. Kreimer. On the hopf algebra structure of perturbative quantum field theories. *Adv. Theor. Math. Phys*, 2(2):303–334, 1998.

[69] G. Kreweras. Sur les partitions non croisées d'un cycle. *Discrete Math*, 1:333–350, 1972.

[70] M. Kuba and A. Panholzer. A unifying approach for proving hook-length formulas for weighted tree families. *Graphs Combin*, 29:1839–1865, 2013.

[71] P. Leroux and B. Miloudi. Généralisations de la formule d'otter. *Ann. Sci. Math. Quebec*, 16(1):53–80, 1992.

[72] A. A. Mahmoud. *An Asymptotic Expansion for the Number of 2-Connected Chord Diagrams*, 2020.

[73] T. Mansour. *Combinatorics of Set Partitions*. Chapman and Hall/CRC, first edition, 2012.

[74] A. Marcus and G. Tardos. Excluded permutation matrices and the Stanley–Wilf conjecture. *J. Combin. Theory Ser. A*, 107(1):153–160, 2004.

[75] N. Marie and K. Yeats. A chord diagram expansion coming from some Dyson-Schwinger equations. *Commun. Number Theory Phys*, 7(2):251–291, 2013.

[76] Jon McCammond. Noncrossing partitions in surprising locations. *American Math. Monthly*, 113(7):598–610, 2006.

[77] K. Menger. Zur allgemeinen kurventheorie. *Fund. Math*, 10:96–115, 1927.

[78] M. M. Murphy and V. R. Vatter. Profile classes and partial well-order for permutations. *Electron. J. Combin*, 9(2), 2003.

[79] L. Nabergall. The combinatorics of a tree-like functional equation for connected chord diagrams, 2021.

[80] W. Naji. Reconnaissance des graphes de cordes. *Discrete Math*, 54:329–337, 1985.

[81] A. Nijenhuis and H. Wilf. The enumeration of connected graphs and linked diagrams. *J. Combin. Theory Ser. A*, 27(3):356–359, 1979.

[82] J. M. Pallo. Right-arm rotation distance between binary trees. *Inform. Process. Lett*, 87:173–177, 2003.

[83] A. Panholzer and H. Prodinger. Level of nodes in increasing trees revisited. *Random Structures Algorithms*, 31(2):203–226, 2007.

[84] E. Panzer. Hopf-algebraic renormalization of Kreimer's toy model. Master's thesis, Humboldt-Universität zu Berlin, 2011.

[85] V. Pilaud and J. Rué. Analytic combinatorics of chord and hyperchord diagrams with k crossings. *Adv. Appl. Math*, 57:60–100, 2014.

[86] C. Pivoteau, B. Salvy, and M. Soria. Algorithms for combinatorial structures: well-founded systems and Newton iterations. *J. Combin. Theory Ser*, 119(8):1711–1773, 2012.

[87] J. Riordan. The distribution of crossings of chords joining pairs of $2n$ points on a circle. *Math. Comp*, 29(129):215–222, 1975.

[88] G. Robinson. On the representations of the symmetric group. *Amer. J. of Math.*, 60(3):745–760, 1938.

[89] L. Rotheray. Hopf subalgebras from green's functions. Master's thesis, Humboldt-Universität zu Berlin, 2015.

[90] B. E. Sagan. Pattern avoidance in set partitions. *Ars Combin*, 94:79–96, 2010.

[91] C. Schensted. Longest increasing and decreasing subsequences. *Canad. J. Math*, 13:179–191, 1961.

[92] R. Simion and F. W. Schmidt. Restricted permutations. *European J. Combin*, 6(4):383–406, 1985.

[93] N. Sloane and The OEIS Foundation Inc. The on-line encyclopedia of integer sequences, 2022.

[94] R. Speicher. Free probability theory and non-crossing partitions. *Sem. Lothar. Combin*, 39, 1997.

[95] R. P. Stanley. *Enumerative combinatorics*, volume 1. Cambridge Univ. Press, Cambridge, second edition, 2012.

[96] R. P. Stanley. Catalan addendum, 2013.

[97] P. R. Stein. On a class of linked diagrams, I. enumeration. *J. Combin. Theory Ser. A*, 24(3):357–366, 1978.

[98] P. R. Stein and C. J. Everett. On a class of linked diagrams, II. asymptotics. *Discrete Math*, 21(3):309–318, 1978.

[99] A. Stoimenow. On the number of chord diagrams. *Discrete Math*, 218(1):209–233, 2000.

[100] D. Tamari. The algebra of bracketings and their enumeration. *Nieuw Arch. Wiskd Ser. 3*, 10:131–146, 1962.

[101] J. Touchard. Sur un probleme de configurations et sur les fractions continues. *Canad. J. Math*, 4:2–25, 1952.

[102] W. T. Tutte. A census of planar triangulations. *Canad. J. Math*, 14:21–38, 1962.

[103] P. van der Laan and I. Moerdijk. Families of Hopf algebras of trees and pre-lie algebras. *Homology, Homotopy Appl*, 8(1):243–256, 2006.

[104] V. Vatter. Enumeration schemes for restricted permutations. *Combin. Probab. Comput*, 17(1):137–159, 2008.

[105] V. Vatter. Finding regular insertion encodings for permutation classes. *J. Symbolic Comput*, 47(3):259–265, 2012.

[106] T. R. S. Walsh and A. B. Lehman. Counting rooted maps by genus III: Nonseparable maps. *J. Combin. Theory Ser. B*, 18:222–259, 1975.

[107] J. West. *Permutations with forbidden subsequences; and, Stack-sortable permutations*. PhD thesis, MIT, 1990.

[108] J. West. Generating trees and forbidden subsequences. *Discrete Math*, 157(1):363–374, 1996.

[109] N. C. Wormald. A correspondence for rooted planar maps. *Ars Combin*, 9:11–28, 1980.

[110] K. Yeats. *Growth estimates for Dyson-Schwinger equations*. PhD thesis, Boston University, 2008.

[111] K. Yeats. *A Combinatorial Perspective on Quantum Field Theory*. Springer, first edition, 2017.

[112] Y. N. Yeh. *On the Combinatorial Species of Joyal*. PhD thesis, State University of New York at Buffalo, 1985.

[113] Y. N. Yeh. The calculus of virtual species and $\mathbb{K}$-species. In *Combinatoire Enumerative*, pages 351–369. Springer-Verlag, 1986.

[114] D. Zeilberger. Enumeration schemes and, more importantly, their automatic generation. *Ann. Comb*, 2(2):185–195, 1998.

APPENDICES

Appendix A

Proof tree for the class $\mathcal{C}(B_{>3})$

Here, we display the indicated proof tree.

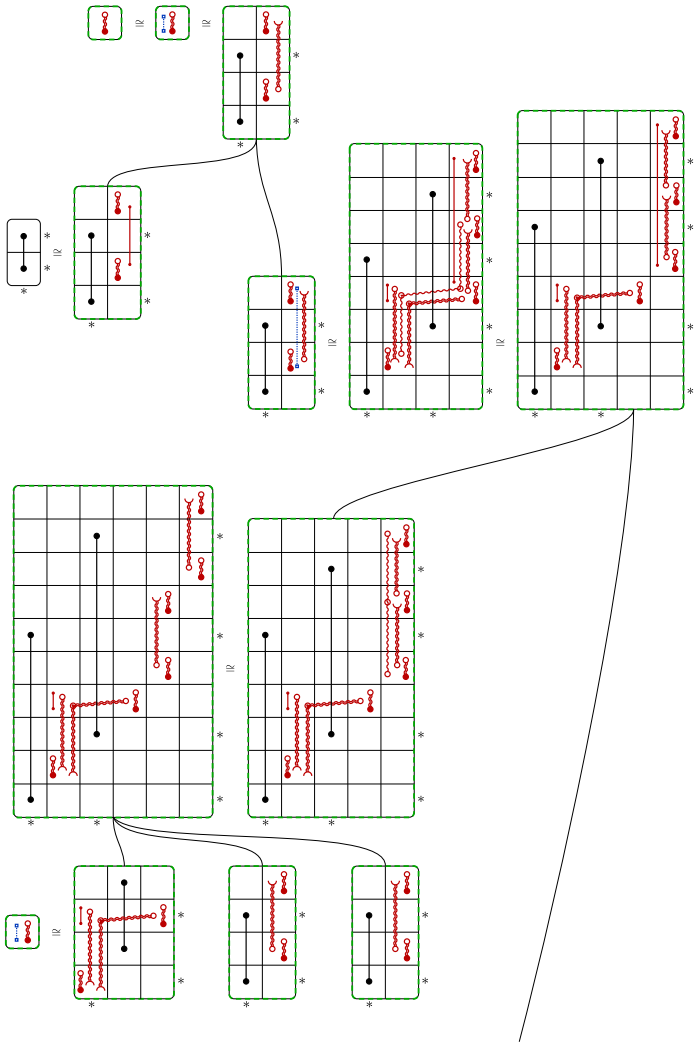


Figure A.1: The first part of a proof tree for the class $\mathcal{C}(B_{>3})$ of connected bottom-cycle-free diagrams.

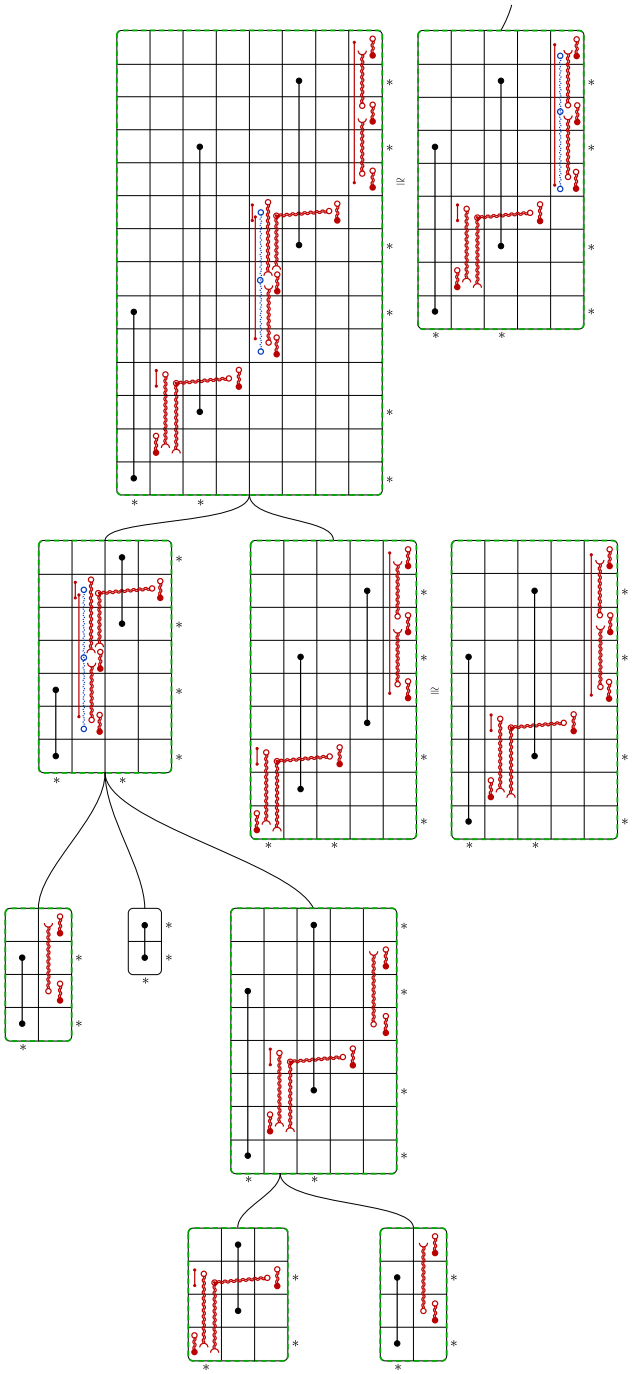


Figure A.1: The second part of a proof tree for the class $\mathcal{C}(B_{>3})$ of connected bottom-cycle-free diagrams.