

# GENERALIZED CHORD DIAGRAM EXPANSIONS OF DYSON-SCHWINGER EQUATIONS

MARKUS HIHN AND KAREN YEATS

ABSTRACT. Series solutions for a large family of single equation Dyson-Schwinger equations are given as expansions over decorated rooted connected chord diagrams. The analytic input to the new expansions are the expansions of the regularized integrals for the primitive graphs building the Dyson-Schwinger equation. Each decorated chord diagram contributes a weighted monomial in the coefficients of the expansions of the primitives and so indexes the analytic solution in a tightly controlled way.

## 1. INTRODUCTION

Dyson-Schwinger equations are integral equations in quantum field theory; they correspond to the classical equations of motion and so are physically highly meaningful and important. On the more mathematical side, Dyson-Schwinger equations have a recursive structure which mirrors the decomposition of Feynman diagrams into subdiagrams. This means that Dyson-Schwinger equations act similarly to functional equations satisfied by generating functions, and so combinatorial tools are useful in understanding them.

A first step in this direction is strictly diagrammatic, and views Dyson-Schwinger equations as equations which recursively generate the Feynman diagrams themselves, or similarly rooted trees representing their subdiagram structure. This view neglects the analytic side – the diagrams still need to be evaluated, but is already interesting, and has been pursued by Loïc Foissy [4, 5].

The next step is to incorporate the analytic information. We would like to do so while maintaining a combinatorial understanding of the objects. In [11] (also available as [10] with an updated final chapter) the second one of us discussed a transformation of a class of analytic Dyson-Schwinger equations into a different form which is well suited for this kind of treatment, see (3). In [7] the second author, along with Nicolas Marie, gave the series solution to a particular, fairly narrow class of such analytic Dyson-Schwinger equations as an expansion indexed by rooted connected chord diagrams, where each chord diagram contributes a single monomial. This expansion was novel and unexpected, hence somewhat interesting. However, it was unclear to what extent it could be generalized to a broader class of Dyson-Schwinger equations, hence it was unclear whether or not this chord diagram expansion was just a peculiarity or how much it might or might not be telling us something actually interesting for physics.

In this paper, which is based on the PhD thesis of the first author [6], we generalize the chord diagram expansion of [7] to a substantially larger class of Dyson-Schwinger equations which includes the form of typical single equation, single scale Dyson-Schwinger equations in

---

MH was supported by the Humboldt Foundation and wants to thank Dirk Kreimer for his help. KY is supported by an NSERC discovery grant.

physics. The main result, Theorem 7.4 is a series solution to any Dyson-Schwinger equation of this class. As in the special case studied before, this expansion is indexed by rooted connected chord diagrams with each contributing a single monomial. The difference is that the chord diagrams are now decorated with the set of possible decorations determined by shape of the Dyson-Schwinger equation, and the monomials come with a weight depending on the chord diagram and its decorations.

The structure of this paper is as follows. First we will briefly discuss the Dyson-Schwinger set up in Section 2. Section 3 defines rooted connected chord diagrams and the features of them which we will need. Section 4 defines the insertion tree of a chord diagram and the weight of a chord diagram. Note that these insertion trees are different from the insertion trees of Feynman graphs which are trees which capture the subdivergence structure of Feynman graphs. Section 5 investigates the insertion trees in more detail answering the question of how the labels of two subtrees can be consistently combined into one tree. Section 6 looks at the decomposition of chord diagrams coming from decomposing the corresponding trees into the two subtrees given by the children of the root. The main result is presented and proved in Section 7, and the paper ends with a brief conclusion.

## 2. DYSON-SCHWINGER EQUATIONS

Suppose we begin with a Dyson-Schwinger equation which is more or less in a recognizable physics form, having only nonstandard notation and normalization, for example

$$(1) \quad G(x, L) = 1 - \frac{x}{q^2} \int d^4k \frac{k \cdot q}{k^2 G(x, \log k^2) (k + q)^2} - \dots \Big|_{q^2 = \mu^2}$$

This is the Dyson-Schwinger equation for the part of the massless fermion self-energy in Yukawa theory which is formed by inserting  $\rightarrow \curvearrowright \rightarrow$  into itself iteratively in all possible ways. In (1)  $x$  is playing the role of the coupling constant,  $q$  is the momentum going through, renormalization is taking place by subtraction at a fixed reference scale  $\mu$ , and  $G(x, L)$  is the fermion Green function. This Dyson-Schwinger equation was solved by Broadhurst and Kreimer in [1].

Now suppose we expand  $G(x, L)$  in  $L$ , convert logarithms to powers using  $\frac{d^k}{d\rho^k} y^\rho|_{\rho=0} = \log^k(y)$ , swap the order of the operators, and recombine the expansion. Then we obtain

$$(2) \quad G(x, L) = 1 - xG(x, \partial_{-\rho})^{-1}(e^{-L\rho} - 1)F(\rho)|_{\rho=0}$$

where  $F(\rho)$  is the Feynman integral for the primitive<sup>1</sup>,  $\rightarrow \curvearrowright \rightarrow$  in this case, regularized on the propagator we are inserting at and the integral evaluated at  $q^2 = 1$ . In this case  $F(\rho) = 1/(\rho)(1 - \rho)$ . Example 3.7 of [11] shows this example in detail.

In [7], (2) is solved as an expansion over rooted connected chord diagrams. However, the form of (2) is quite specialized. First of all there is only one primitive Feynman graph. Second  $G(x, L)$  appears once in the denominator of the integrand of the original Dyson-Schwinger equation. This corresponds to the fact that there is one insertion place. The

---

<sup>1</sup>This Feynman graph is primitive in the appropriate renormalization Hopf algebra, hence the term *primitive*. This algebraic framework, while it underpins the entire theory to which this paper contributes, will not be used directly and so will not be defined. The reader can see [2, 3, 8, 9] to read more about it.

more general equation which we will solve here is

$$(3) \quad G(x, L) = 1 - \sum_{k \geq 1} x^k G(x, \partial_{-\rho})^{1-sk} (e^{-L\rho} - 1) F_k(\rho)|_{\rho=0}$$

In this equation,  $k$  indexes the loop numbers of the primitives.  $s$  is a parameter indicating the degree of the insertion. The  $F_k(\rho)$  are again the regularized Feynman integrals of the primitives. From now on we will view the  $F_k(\rho)$  as given. Since we are interested in series solutions we will always think of them in terms of their expansions which we assume to have a simple pole at 0. We will use the notation

$$F_k(\rho) = \sum_{i \geq 0} a_{k,i} \rho^{i-1}$$

for the expansions of the  $F_k$ .

The previous example, (2), was the  $k = 1, s = 2$  case of (3). The photon in quantum electrodynamics would be an  $s = 1$  case. We can see this because at 1 loop ( $k = 1$ ) there is no insertion place for a photon in the one loop photon correction, for any two loop photon correction there is one insertion place for a photon, and the sequence continues as described above. See [11] Section 3.3.2 for a discussion of the combinatorics of counting insertions.

Returning to the idea of combinatorial Dyson-Schwinger equations mentioned in the introduction, a good way to think about Dyson-Schwinger equations as in (3), is to first have a combinatorial Dyson-Schwinger equation in one variable  $x$ , which captures how the contributing Feynman diagrams (or trees) are formed recursively. Then the combinatorial Dyson-Schwinger equation can be directly translated into the form of (3) which we call the associated analytic Dyson-Schwinger equation. See [11] for details.

From a mathematical perspective the problem we will solve in this paper is the following. Suppose  $s$  and the expansions of the  $F_k$  are given. The Dyson-Schwinger equation (3) recursively defines the coefficients of  $G(x, L)$  viewed as a bivariate power series in  $x$  and  $L$ . However, it does not do so in a nice or easy to use way. We will give an explicit, combinatorial description of the series  $G(x, L)$  solving (3), see Theorem 7.4. This gives the solution to this substantial class of Dyson-Schwinger equations as a kind of weighted generating function of chord diagrams.

There is one property of the series expansion of  $G(x, L)$  which we can see directly from the Dyson-Schwinger equation. This is renormalization group equation translated into this context.

**Proposition 2.1.** *Let  $G(x, L)$  satisfy (3). Suppose*

$$G(x, L) = 1 - \sum_{i \geq 1} L^i \gamma_i(x) \quad \text{and} \quad \gamma_i(x) = \sum_{j \geq i} \gamma_{i,j} x^j$$

*Then*

$$\gamma_k(x) = \frac{1}{k} \gamma_1(x) \left( 1 - sx \frac{d}{dx} \right) \gamma_{k-1}(x)$$

*Proof.* Since  $G$  satisfies (3) it satisfies the renormalization group equation. Extracting coefficients of  $L$  gives the result. The result can also be proved algebraically by using the Dynkin operator  $S \star Y$ . See [11] or [10] chapter 4 for details of both approaches.  $\square$

### 3. CHORD DIAGRAMS

**Definition 3.1.** A rooted chord diagram  $D$  of size  $n$  is a fixed point free involution  $D \in S_{2n}$ ; that is, a permutation such that  $D^2 = id_{S_{2n}}$  with  $D(i) \neq i$  for all  $i = 1 \dots 2n$ . Equivalently  $D$  is a permutation that can be written as product of disjoint transpositions without fixed points:

$$D = (x_1y_1)(x_2y_2) \cdots (x_ny_n)$$

where  $x_1 < \dots < x_n$  and  $x_i < y_i$  for all  $i = 1 \dots 2n$ . Each transposition is called a chord and  $(x_1y_1)$  is called the root chord.

We can visualize a rooted chord diagram as follows. Draw a circle and mark  $2n$  vertices on it. Choose a distinct vertex as the root and label that vertex 1. Enumerate the vertices counter-clockwise and draw a chord between vertex  $x_i$  and vertex  $y_i$  for each transposition.

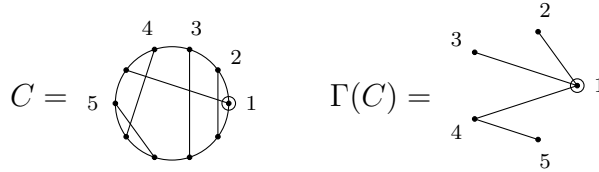
We say that a chord  $(x_iy_i)$  crosses a chord  $(x_jy_j)$  if

$$x_i < x_j < y_i < y_j \text{ or } x_j < x_i < y_j < y_i$$

**Definition 3.2.** Let  $C$  be a rooted chord diagram.

- The intersection graph of  $C$  is the graph with a vertex for each chord, where the vertex is labeled  $i$  for chord  $(x_iy_i)$ , and with two vertices joined by an edge if the corresponding chords cross.
- $C$  is connected if its intersection graph is connected
- The terminal chords of  $C$  are those chords  $(x_i, y_i)$  which do not cross any chords  $(x_j, y_j)$  with  $i < j$ . The set of terminal chords of  $C$  is denoted  $ter(C)$ .

**Example 3.3.** A rooted connected chord diagram  $C$  with its intersection graph  $\Gamma(C)$ :



We will be interested in rooted connected chord diagrams where each chord is assigned a decoration from a set of possible decorations. Such a chord diagram is called a decorated chord diagram.

**Definition 3.4.**

- Let  $\mathcal{R}$  be the set of rooted connected chord diagrams.
- Let  $\mathcal{R}_n$  be the set of rooted connected chord diagrams with  $n$  chords.
- Let  $\mathcal{R}_{dec}$  be the set of all decorated rooted connected chord diagrams with the chord decorations from a set  $D$ .
- If  $c$  is a chord of a decorated chord diagram we will write  $d(c)$  for the decoration of  $c$ .

The order we need for our constructions is not the obvious counterclockwise order, but rather a different order which we define recursively as follows.

**Definition 3.5** (Intersection order). The intersection order of a rooted chord diagram  $C$  is defined recursively by the following pseudo code:

```

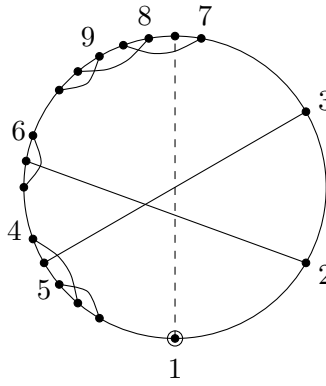
intersection_order(k, C) {
  m := root(C)
  label(m) := k
  k := k + 1
  if |C| != 1 then
  foreach D := connected_components(C \ m) traversed counter clockwise
  {
    intersection_order(k,D)
    k := k + |D|
  }
}

```

and start the procedure with

`intersection_order(1,C)`.

The following picture shows a chord diagram where its chords are labeled in the intersection order. The root chord is indicated with a dotted line.



From now on always assume chords are labeled in intersection order.

**Definition 3.6.** Let  $Ter(C) = \{t_0 < \dots < t_n\}$  and  $d_k$  be the decoration of the  $k$ -th chord, then we define:

$$\|C\| := \sum_{c=1}^{|C|} d_c$$

$$\hat{a}_C := \left( \prod_{c=1}^n a_{d_{t_c}, t_c - t_{c-1}} \right) \cdot \left( \prod_{k \notin ter(C)} a_{d_k, 0} \right)$$

The symbol  $b(C)$  denotes the label of the base chord which is defined to be the smallest terminal chord.

Note that the hat notation in the above definition does not denote an operator but only that the monomial does not contain  $a_{d_{b(C)}, b(C)-k}$ .

**Definition 3.7** (Insertion operation for rooted chord diagrams). *Let  $C \in \mathcal{R}(n), D \in \mathcal{R}(m)$  and let*

$$\begin{aligned} C &= (x_1 y_1) \dots (x_n y_n) \\ D &= (x'_1 y'_1) \dots (x'_m y'_m) \end{aligned}$$

*their underlying permutations. Then for each  $k = 1, \dots, 2m - 1$  define  $C \circ_k D \in \mathcal{R}(n + m)$  by the following permutation:*

$$(x_1, y_1 + k) \dots (x_n + k, y_n + k) (H_{n,k}(x'_1), H_{n,k}(y'_1)) \dots (H_{n,k}(x'_m), H_{n,k}(y'_m))$$

*where  $H_{n,k}$  is defined to be:*

$$H_{n,k}(x) = \begin{cases} x + 1 & \text{if } x \leq k \\ x + 2n & \text{otherwise} \end{cases}$$

Intuitively what this insertion operation does is put the root of  $C$  immediately before the root of  $D$  and then put the rest of  $C$  into the  $k$ th interval between the ends of the chords of  $D$ .

$C \circ_k D$  is indeed a rooted connected chord diagram of size  $n + m$ , because every integer from one to  $2(n + m)$  appears exactly once in the transpositions, it is fix point free and the labeling shift does not destroy any crossings but the root chord of  $C$  creates at least one new crossing with chords of  $D$ .

**Example 3.8.**

$$(1, 4)(2, 6)(3, 5) \circ_2 (1, 4)(2, 5)(3, 6) = (1, 6)(2, 10)(3, 11)(4, 8)(5, 7)(9, 12)$$

**Remark 3.9.** Note that the insertion operation is highly non associative and non commutative. If  $C \circ_k D$  is defined,  $D \circ_k C$  may not be defined. For example, if  $C$  is a chord diagram with only one chord, then  $D \circ_k C$  is not defined for  $k \geq 2$  because there is only one insertion interval in  $C$  by definition.

In the following definition of the root share decomposition, we will need to decompose chord diagrams. Since by our definition chord diagrams are certain permutations, we need to define for an expression

$$A = (a_1 a_2) \dots (a_{2n-1} a_{2n})$$

where  $a_k$  ( $k = 1..2n$ ) are arbitrary distinct natural numbers, the associated normalized involution norm( $A$ ) by

$$\text{norm}(A) = (\sigma(1)\sigma(2)) \dots (\sigma(2n-1)\sigma(2n))$$

where  $\sigma \in S_{2n} : a_{\sigma^{-1}(1)} < a_{\sigma^{-1}(2)} \dots < a_{\sigma^{-1}(2n)}$ .

**Example 3.10.** Let  $A = (13)(28)(57)$ , then  $\text{norm}(A) = (13)(26)(45)$

**Definition 3.11** (Root share decomposition). *Let  $C \in \mathcal{R}$  with  $|C| > 1$ , there exists a unique  $i$  such that*

$$C = C' \circ_i C'' \text{ where } C' = \text{norm}(C \setminus C_1), C'' = \text{norm}(C_1)$$

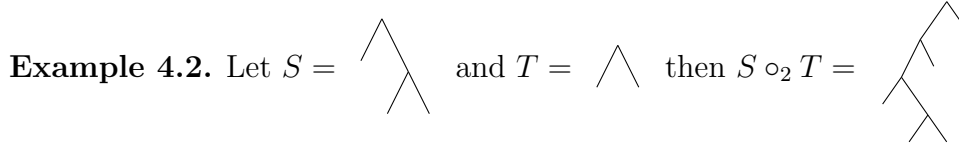
*and  $C_1$  is the first connected component of  $C$  with the root chord removed. Note that  $C \setminus C_1$  is always connected. This decomposition is called the root share decomposition of  $C$ .*

#### 4. INSERTION TREES AND WEIGHTS

We will now need to associate a binary rooted tree to each rooted connected chord diagram. The leaves of the tree will correspond to the chords and will be labeled accordingly. This same construction was used in [7], however, there the construction appeared to only be a technical tool to prove a certain recurrence. In the more generalized case discussed here, the tree is used to define the weight with which each chord diagram contributes to the Green function. Thus, the rooted trees are not merely technical, but actually capture a key part of the structure which we need in the general case.

**Definition 4.1** (Insertion operation on rooted plane trees). *Let  $T, T'$  be rooted plane trees with a virtual edge above the root. Label the virtual edge 1 and label the remaining edges following a pre-order traversal. The rooted plane tree  $T \circ_k T'$  is defined by*

- putting a new vertex in the middle of edge  $k$  of  $T'$ ,
- placing  $T$  as the left subtree rooted at this vertex and
- placing the subtree of  $T'$  rooted at the bottom end of  $k$  as the right subtree of the new vertex.



Recall that chord diagrams are labeled by the intersection order.

**Definition 4.3** (The binary tree  $T(C)$  associated to  $C$ ). *Let  $C = C_1 \circ_k C_2$  decomposed by the root share decomposition and let  $C_1, C_2$  be labeled by the induced labeling of  $C$ . Then  $T(C)$  is defined recursively by*

$$T(C) = \begin{cases} \text{one vertex labeled by } l & \text{if } |C| = 1 \text{ and is labeled by } l \text{ in the induced labeling} \\ T(C_1) \circ_k T(C_2) & \text{when } C = C_1 \circ_k C_2 \end{cases}$$

**Definition 4.4.** *Let  $v$  be a leaf of a binary rooted tree. Consider the path beginning at  $v$  and moving up and to the left as long as such an edge exists. Define  $\nu_v$  to be the number of edges in this path.*

Note that if a leaf is a left child then  $\nu_v = 0$ . For a more precise definition of  $\nu_v$  using the binary string representation of a binary rooted tree see [6].

**Definition 4.5** (Branch-left vector). *Let  $C$  be a rooted, connected chord diagram of size  $n$  and  $T(C)$  its corresponding unique insertion tree, then  $\nu(C) = (\nu_1, \dots, \nu_n)$  is said to be the branch-left vector of  $C$ .*

Now, we can define the weight mentioned in the introduction of this section.

**Definition 4.6** (The weight of a decorated rooted connected chord diagram). *For a chord diagram  $C \in R_{dec}$  with branch-left vector  $\nu(C)$  and decoration  $d_i$  for chord  $i$  define*

$$\omega(C) = \prod_{k=1}^{|C|} \binom{d_k s + \nu_k(C) - 2}{\nu_k(C)}$$

where  $s$  is the parameter given by our Dyson-Schwinger equation. Further, we denote by  $\omega_{\hat{a}_C}$  the weight associated to  $C$  but without the factor corresponding to its base chord  $b(C)$ .

Whenever convenient we will write  $\omega_C$  instead of  $\omega(C)$ .

Our first use of the weights will be to show that the renormalization group equation holds for the expansion over chord diagrams which will solve the Dyson-Schwinger equation. To keep track of the two expansions – the expansion of the Dyson-Schwinger equation itself and the chord diagram expansion – which will ultimately prove to be the same, we will distinguish them by superscripts: comb for the combinatorial expansion, dif for the analytic expansion. Specifically, Let

$$G^{\text{dif}}(x, L) = 1 - \sum_{k=0}^N x^k G^{\text{dif}}(x, \partial_{-\rho=0})^{1-sk} (e^{-L\rho} - 1) F_k(\rho)$$

and write

$$g_k^{\text{dif}}(x) = \frac{(-1)^k}{k!} [L^k] G^{\text{dif}}(x, L)$$

On the combinatorial side define

$$g_k^{\text{comb}}(x) = \sum_{\substack{C \in \mathcal{R}^{\text{dec}} \\ b(C) \geq k}} x^{|C|} \omega_C \hat{a}_C a_{d_b(C), b(C)-k}$$

and  $G^{\text{comb}}$  as the analogous sum of the  $g^{\text{comb}}$ .

By Proposition 2.1 we already know  $g^{\text{dif}}$  satisfies the renormalization group equation. Next we show the analogous result for  $g^{\text{comb}}$ .

**Theorem 4.7** (Renormalization group equation for  $g^{\text{comb}}$ ).

$$g_k^{\text{comb}}(x) = g_1^{\text{comb}}(x) \cdot (sx\partial_x - 1)g_{k-1}^{\text{comb}}(x)$$

To prove this theorem we need to answer the following two questions:

- (1) How is the monomial of a chord diagram  $C$  recovered from the root share decomposition  $C = C_1 \circ_r C_2$ ?
- (2) How is the weight of a chord diagram  $C$  recovered from the root share decomposition  $C = C_1 \circ_r C_2$ ?

The following two lemmas answer these questions and together they are enough to prove Theorem 4.7. The monomial associated to a decorated chord diagram  $C \in \mathcal{R}_{\text{dec}}$  with root share decomposition  $C = C_1 \circ_k C_2$  can be reconstructed from  $C_1, C_2$  in the following sense:

**Lemma 4.8** (RSD monomial Lemma). *Let  $C_1, C_2 \in \mathcal{R}_{\text{dec}}$  with  $C = C_1 \circ_k C_2$  and  $d, d_1, d_2$  the corresponding decorations of the base chords, i.e.  $d := d(b(C)), d_1 := d(b(C_1)), d_2 := d(b(C_2))$ , then*

$$\hat{a}_C a_{d, b(C)-l} = \hat{a}_{C_1} a_{d_1, b(C_1)-1} \hat{a}_{C_2} a_{d_2, b(C_2)-l+1}$$

where  $1 < l < b(C)$

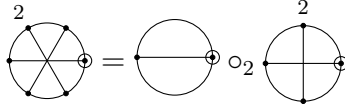
*Proof.* The proof is the same as the proof of Lemma 4.2 in [7] but keeping track of decorations.  $\square$



**Example 4.9.** Let  $C$  be the wheel with three spokes as a rooted connected chord diagram and choose as decoration a two for the last chord (only the non-trivial decoration is included in the pictures):



As calculated earlier  $C = C_1 \circ_2 C_2$



The left hand side of the previous Lemma is:

$$\hat{a}_C a_{d,b(C)-l} = a_{1,0}^2 a_{2,3-l}$$

The right hand side of the previous Lemma contains

$$\begin{aligned} \hat{a}_{C_1} &= 1 \\ a_{d_1,b(C_1)-l} &= a_{1,0} \\ \hat{a}_{C_2} &= a_{1,0} \\ a_{d_2,b(C_2)-l+1} &= a_{2,2-l+1} \end{aligned}$$

which agrees with the lemma.

Returning to the general case, we need to understand how the root share decomposition relates to the branch left vectors. Consider a chord diagram and its root share decomposition  $C = C' \circ_k C''$ . We know that branch-left vector  $\nu(C')$  is copied into  $C$  so only the branch left vector of  $C''$  is modified. This yields the following equation:

$$\sum_{k=1}^{2n-1} \omega(C' \circ_k C'') = \omega(C') \sum_{k=1}^{2n-1} \omega({}^{\circ_k} C'')$$

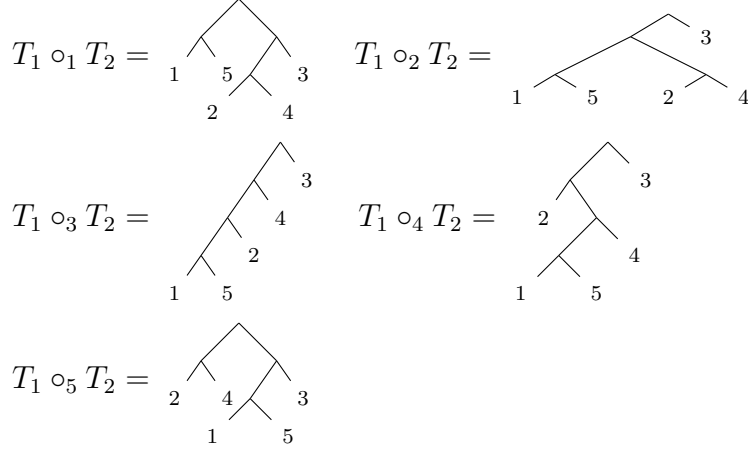
where  ${}^{\circ_k}$  is defined as follows:

**Definition 4.10** (Virtual insertion  ${}^{\circ_k}$ ). *Let  $C \in \mathcal{R}_{dec}$  then  ${}^{\circ_k} C$  is defined to be the same chord diagram but with modified tree:  $T({}^{\circ_k} C)$  is  $T(C)$  but with an additional vertex  $v$  and an additional left child inserted before the  $k$ -th vertex  $w$ . As a result  $w$  will be the right child of  $v$ .*

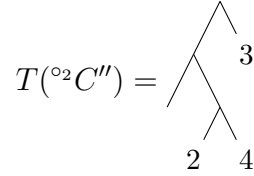
The following example illustrates this definition.

**Example 4.11.** Let  $C' = \textcircled{+}$  and  $C'' = \textcircled{+}$  and the associated trees with the induced

labeling are:  $T_1 := T(C') = \begin{array}{c} \wedge \\ 1 \quad 5 \end{array}$  and  $T_2 := T(C'') = \begin{array}{c} \wedge \\ 2 \quad 4 \end{array} \begin{array}{c} \wedge \\ \quad 3 \end{array}$ .



Note that the branch left vector of  $C'$  is never changed, so we can replace it as a marker vertex. This is what the virtual insertion does:



**Lemma 4.12.** Let  $C', C''$  be decorated chord diagrams where  $|C''| = n$ , then:

$$\sum_{k=1}^{2n-1} \omega(C' \circ_k C'') = \omega(C') \omega(C'') (s\|C''\| - 1)$$

*Proof.* Note that  $C' \circ_k C''$  does not affect the tree form of  $C'$  in any way so we get

$$\sum_{k=1}^{2n-1} \omega(C' \circ_k C'') = \omega(C') \sum_{k=1}^{2n-1} \omega(^{\circ k} C'')$$

Now notice that there are  $\nu_k + 1$  possibilities to increase the left branch by 1.

$$\sum_{k=1}^{2n-1} \omega(^{\circ k} C'') = (\nu_1 + 1) \omega^{+1}(C'') + \dots + (\nu_n + 1) \omega^{+n}(C'')$$

where  $\omega^{+k}(C'')$  is defined as the weight of  $C''$  after incrementing the  $k$ -th component of the branch left vector:

$$\omega^{+k}(C'') = \omega(C'') \left( 1 + \frac{sd_k - 2}{\nu_k + 1} \right)$$

Plugging this into the latter equation, we get the result:

$$\begin{aligned} \sum_{k=1}^{2n-1} \omega({}^{\circ k} C'') &= \omega(C'') \sum_{k=1}^n (\nu_k + 1 + s d_k - 2) \\ &= \omega(C'') \left( n - 1 - n + s \sum_{k=1}^n d_k \right) = \omega(C'') (s \|C''\| - 1) \end{aligned}$$

□

**Example 4.13.** Consider the following decorated chord diagrams (the decorated chords are thickened and the decoration is on the other side from the labeling).

$$C' = \begin{array}{c} 2 \\ \bigcirc \\ \oplus 1 \\ d_1 \end{array} \quad C'' = \begin{array}{c} 2 \\ \bigcirc \\ \oplus 1 \\ d_2 \end{array}$$

Clearly,  $\nu(C') = \nu(C'') = (0, 1)$  and thus we have

$$\begin{aligned} \omega(C') &= d_1 s - 1 & \omega(C'') &= d_2 s - 1 \\ \omega(C') \omega(C'') (s \|C''\| - 1) &= (d_1 s - 1)(d_2 s - 1)(s(d_2 + 1) - 1) \end{aligned}$$

The branch-left vectors for the different insertions are:

$$\nu(C' \circ_1 C'') = (0, 1, 0, 2) \quad \nu(C' \circ_2 C'') = (0, 1, 1, 1) \quad \nu(C' \circ_3 C'') = (0, 1, 0, 2)$$

Thus, for the the sum of the left hand side of the lemma we have:

$$\begin{aligned} \omega(C' \circ_1 C'') &= \omega(C' \circ_3 C'') = \binom{d_2 s}{2} (d_1 s - 1) \\ \omega(C' \circ_2 C'') &= (d_1 s - 1)(d_2 s - 1)(s - 1) \end{aligned}$$

which is what the Lemma 4.12 tells us it should be.

*Proof of theorem 4.7.* : To prove

$$g_k^{\text{comb}}(x) = g_1^{\text{comb}}(x) \cdot (sx \partial_x - 1) g_{k-1}^{\text{comb}}(x),$$

let us do the differential first:

$$(sx \partial_x - 1) g_{k-1}^{\text{comb}}(x) = \sum_{\substack{C \in \mathcal{R}^{\text{dec}} \\ b(C) \geq k-1}} (s \|C\| - 1) x^{\|C\|} \omega_C \hat{a}_C a_{b(C)-k+1}$$

Multiplying  $g_1^{\text{comb}}$  from the left we obtain:

$$\begin{aligned} g_1^{\text{comb}}(x) \cdot (sx \partial_x - 1) g_{k-1}^{\text{comb}}(x) &= \\ \left[ \sum_{\substack{C' \in \mathcal{R}^{\text{dec}} \\ b(C') \geq 1}} x^{\|C'\|} \omega_{C'} \hat{a}_{C'} a_{b(C')-1} \right] &\left[ \sum_{\substack{C'' \in \mathcal{R}^{\text{dec}} \\ b(C'') \geq k-1}} (s \|C''\| - 1) x^{\|C''\|} \omega_{C''} \hat{a}_{C''} a_{b(C'')-k+1} \right] = \\ \sum_{\substack{C' \in \mathcal{R}^{\text{dec}}, C'' \in \mathcal{R}^{\text{dec}} \\ b(C') \geq 1, b(C'') \geq k-1}} x^{\|C'\| + \|C''\|} \omega_{C'} \omega_{C''} (s \|C''\| - 1) &\hat{a}_{C'} a_{b(C')-1} \hat{a}_{C''} a_{b(C'')-k+1} \end{aligned}$$

By Lemma 4.8 and Lemma 4.12 the result follows.  $\square$

## 5. SHUFFLING TREES

The next thing we need to understand is, if we have two rooted connected chord diagrams  $D_1$  and  $D_2$  with corresponding trees  $H_1$  and  $H_2$ , what possible chord diagrams can correspond to the binary rooted tree with left child  $H_1$  and right child  $H_2$ . The relative order of the labels of  $H_1$  and  $H_2$  must remain the same, so it is a question of which shuffles of the labels give trees which correspond to chord diagrams.

In the middle of the shuffling process we will need to consider the original labels of  $H_1$  and  $H_2$  as well as the new labels generated so far. To keep track of this we will use the notation  $\underline{\mathbb{N}}$  and  $\overline{\mathbb{N}}$ .  $\underline{\mathbb{N}}$  will hold the labels for the left tree and  $\overline{\mathbb{N}}$  will hold the labels for the right tree at the start of the labeling procedure. In the labeling procedure elements of  $\underline{\mathbb{N}}$  resp.  $\overline{\mathbb{N}}$  will be successively replaced by the final label elements which will be elements of  $\mathbb{N}$ . Let  $\leq$  resp.  $\overline{\leq}$  be the strict ordering of  $\underline{\mathbb{N}}$  resp.  $\overline{\mathbb{N}}$ . Note that  $\leq$  and  $\overline{\leq}$  are not defined to compare an element of  $\underline{\mathbb{N}}$  with an element of  $\overline{\mathbb{N}}$  and vice versa. However, due to the iterative nature of the algorithm we will consider elements of  $\mathbb{N}$  which are assigned so far to be smaller than every element of the remaining labels from  $\underline{\mathbb{N}} \cup \overline{\mathbb{N}}$ .

Let's define the shuffle product for the special case of two subsets  $\{\underline{1}, \dots, \underline{k}\} \subset \underline{\mathbb{N}}$ ,  $\{\overline{1}, \dots, \overline{l}\} \subset \overline{\mathbb{N}}$ :

$$\{\underline{1}, \dots, \underline{k}\} \sqcup \{\overline{1}, \dots, \overline{l}\} := \left\{ (w_1, \dots, w_{k+l}) : \begin{array}{l} \{w_1, \dots, w_{k+l}\} = \{\underline{1}, \dots, \underline{k}\} \cup \{\overline{1}, \dots, \overline{l}\} \\ \text{and } r < s \Rightarrow w_r \leq w_s \text{ if } w_r, w_s \in \underline{\mathbb{N}} \\ \text{and } r < s \Rightarrow w_r \overline{\leq} w_s \text{ if } w_r, w_s \in \overline{\mathbb{N}} \end{array} \right\}$$

**Example 5.1.**

$$\{\underline{1}, \underline{2}\} \sqcup \{\overline{1}, \overline{2}\} = \left\{ (\underline{1}, \underline{2}, \overline{1}, \overline{2}), (\underline{1}, \overline{1}, \underline{2}, \overline{2}), (\underline{1}, \overline{1}, \overline{2}, \underline{2}), (\overline{1}, \underline{1}, \underline{2}, \overline{2}), (\overline{1}, \underline{1}, \overline{2}, \underline{2}), (\overline{1}, \overline{2}, \underline{1}, \underline{2}) \right\}$$

**Definition 5.2** (Pre-labeling). *Let  $L \subset \mathbb{N} \cup \underline{\mathbb{N}} \cup \overline{\mathbb{N}}$  be a finite set. We call a bijection*

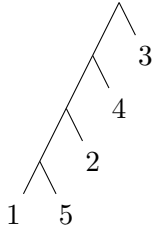
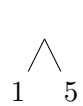

$$\sigma' : \Lambda(T) \rightarrow L$$

*a pre-labeling for  $T$  if the image of  $\sigma'$  contains elements of  $\underline{\mathbb{N}}$  or  $\overline{\mathbb{N}}$ .*

In order to use Proposition 5.6 on pre-labeled trees, we will need the notion of smallest removable subtree containing 1 from Definition 4.6 of [7] in a slightly more general setting. First we define what we mean by removing a subtree from a tree.

**Definition 5.3** (Removing a subtree). *Removing a subtree  $S \subset T$  rooted at a vertex  $w$ , denoted by  $T \setminus S$ , is defined by the following procedure:*

- (1) *Every edge and vertex from  $S$  will be removed from  $T$ .*
- (2) *The edge  $(w, w')$  where  $w \in S$  and  $w' \notin S$  is removed. The edge  $(w', w'')$  outgoing from  $w'$  where  $w'' \notin S$  is contracted.*

**Example 5.4.** Let  $T =$   and  $S =$  , then  $T \setminus S =$  

**Definition 5.5** (Smallest removable subtree). *Let  $(T, \sigma)$  be a rooted, plane, leaf labeled, binary tree  $T$  with a (pre-)labeling  $\sigma$ . A smallest removable subtree  $S$  of  $T$  is defined to be the smallest tree such that  $T \setminus S$  maintains P1 of Proposition 5.6.*

In [7] a complete characterization of labelings of binary trees coming from chord diagrams is given. Specifically:

**Proposition 5.6.** *Let  $\mathcal{T}_n$  be the set of rooted, plane, leaf labeled, binary trees with  $n$  leaves such that for every  $(T, \sigma) \in \mathcal{T}_n$  the following two properties hold*

- P1) *At any vertex  $v$  that is not a leaf the smallest label in the left subtree of  $v$  is smaller than the label at the end of the fully right branch of the right subtree.*
- P2) *Let  $H$  be the smallest removable subtree of  $T$  containing 1.  $H$  contains exactly the following leaf labels:*

$$\text{Im}(\sigma|_H) = \left\{ 1, l(T) - l(H) + 2, l(T) - l(H) + 3, \dots, l(T) \right\}$$

where  $l(\cdot)$  denotes here the maximal label of a tree. Note that  $H$  is the left side of the root share decomposition of trees.

Furthermore, P1 and P2 must stay true recursively in the following sense. Let  $T = H \circ_r (T \setminus H)$  for some  $r$  then P1 and P2 must hold for  $T \setminus H$ . Then every  $(T, \sigma) \in \mathcal{T}_n$  represents a unique rooted connected chord diagram of size  $n$ , so

$$\mathcal{T}_n = \{T(C) : C \in \mathcal{R}, |C| = n\}$$

*Proof.* See [7] □

Call a labeling satisfying these properties admissible.

Let  $k = 1, \dots, |D_1|$  and  $m := b(D_2)$  the base chord corresponding to  $D_2 = \mathcal{T}^{-1}(H_2)$ ,  $\underline{1}, \dots, \underline{n} \in \underline{\mathbb{N}}$  the pre-labeling for  $H_1$  and  $\bar{1}, \dots, \bar{h} \in \bar{\mathbb{N}}$  the pre-labeling for  $H_2$ . The following procedure assigns to a shuffle

$$w = (w_1 \dots w_{k+m}) \in \{\underline{1}, \dots, \underline{k}\} \sqcup \{\bar{1}, \dots, \bar{m}\}$$

an admissible labeling  $\sigma = \sigma(w)$  for the tree with  $H_1$  and  $H_2$  the children of the root. Because this tree will be well defined, we call it  $H_1 \diamond_\sigma H_2$ , see Definition 6.1. The set of shuffles  $\{\underline{1}, \dots, \underline{k}\} \sqcup \{\bar{1}, \dots, \bar{m}\}$  is therefore the set of admissible shuffles associated to  $D_1, D_2$  or equivalently to  $H_1, H_2$  and will be denoted by  $D_1 \sqcup D_2$  resp.  $H_1 \sqcup H_2$ .

**Proposition 5.7.** *Let  $w$  be a shuffle of  $\{\underline{1}, \dots, \underline{k}\}$  and  $\{\bar{1}, \dots, \bar{m}\}$  and  $(H_1, \sigma_{H_1}), (H_2, \sigma_{H_2})$  as before, then the following algorithm produces an admissible labeling  $\sigma$  and a unique leaf labeled tree  $(T, \sigma) \in \mathcal{T}$ :*

- (1) Graft the left and right tree  $H_1$  and  $H_2$  at a new root, merge the pre-labelings and call this tree  $(T_1, \sigma_1)$ . To be more specific:  $T_1 = B_+(H_1 H_2)$  and

$$\sigma_1 : \Lambda(H_1) \cup \Lambda(H_2) \rightarrow \{\underline{1}, \dots, \underline{n}\} \cup \{\bar{1}, \dots, \bar{h}\}$$

is given by

$$\sigma_1(\lambda) = \begin{cases} \sigma_{H_1}(\lambda) & \text{if } \lambda \in H_1 \\ \sigma_{H_2}(\lambda) & \text{otherwise} \end{cases}$$

- (2) For each  $l = 1, \dots, k + m$ , replace the pre-label  $\omega_l$  by the label  $l \in \mathbb{N}$ , i.e. modify  $\sigma_1$  such that  $\sigma_1(\omega_l) = l$
- (3) Assign the label  $b(D_2) + k \in \mathbb{N}$  to the fully right branch leaf of  $T_1$ , i.e. modify  $\sigma_1$  such that

$$\sigma_1(\lambda) = b(D_2) + k$$

where  $\lambda$  is the leaf of the fully right branch of  $T_1$ .

(4) Apply **LABEL**( $T_1, \sigma_1, b(D_2) + k + 1$ ). The labeling procedure **LABEL** is defined as follows:

```

LABEL( $T, \text{ref } \sigma, \text{ref } l$ ) {
  if  $\sigma$  is an admissible label {
    return ( $T, \sigma$ )
  }
   $s := 0$ 
  if  $\text{Im}(\sigma) \subset \mathbb{N} \cup \underline{\mathbb{N}}$  or  $\text{Im}(\sigma) \subset \mathbb{N} \cup \bar{\mathbb{N}}$  {
    // replace the pre-label elements by the next labels  $l$ 
    // in the order that is induced by  $\leq$  resp.  $\bar{<}$ 
    if  $\text{Im}(\sigma) \subset \mathbb{N} \cup \underline{\mathbb{N}}$  {
       $s := |\underline{\mathbb{N}} \cap \text{Im}(\sigma)|$ 
      Let  $\{\underline{\lambda}_1 \leq \dots \leq \underline{\lambda}_s\} = \underline{\mathbb{N}} \cap \text{Im}(\sigma)$ 
      for  $i = 1 \dots s$  {
        replace pre-label  $\underline{\lambda}_i$  by  $l + i$ 
      }
    }
    if  $\text{Im}(\sigma) \subset \mathbb{N} \cup \bar{\mathbb{N}}$  {
       $s := |\bar{\mathbb{N}} \cap \text{Im}(\sigma)|$ 
      Let  $\{\bar{\lambda}_1 \bar{<} \dots \bar{<} \bar{\lambda}_s\} = \bar{\mathbb{N}} \cap \text{Im}(\sigma)$ 
      for  $i = 1 \dots s$  {
        replace pre-label  $\bar{\lambda}_i$  by  $l + i$ 
      }
    }
  }
  // using the extended definition
  // of smallest removable subtree to get
  // the root share decomposition on
  // the level of trees
   $T = T' \circ_r T''$ 
  LABEL( $T'', \sigma, l + s$ )
  LABEL( $T', \sigma, l + s$ )
}

```

Note that **LABEL** does not change the form of  $T$ .

*Proof.* This is the content of Lemma 4.12 of [7] with the shuffling algorithm clarified. The proof is the same.  $\square$

## 6. DIAMOND DECOMPOSITION

Now let us consider the decomposition mentioned in the previous section in more detail. When starting with a tree  $T(C)$  associated to a chord diagram, by removing the root we obtain a left and a right tree that define chord diagrams by themselves. Let us call those chord diagrams  $D_1$  and  $D_2$ , They are well defined for every chord diagram  $C$  and so we

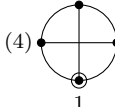
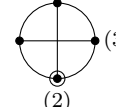
are able to define the diamond operation on chord diagrams. This operation, which will be defined in detail in Definition 6.1, is needed for some technical lemmas that we need to prove the main theorem. The lemmas roughly say that summing over a set of chord diagrams of fixed size  $n$  is the same as summing over all possible decompositions of  $C$  into  $D_1, D_2$ . To be more specific we will need to prove:

$$(4) \quad \sum_{\substack{\|C\|=i+1 \\ b(C)=j+1}} \omega_{\hat{C}} \hat{a}_C = \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} \left( \sum_{\substack{\|D_1\|=k \\ b(D_1) \geq l}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-l} \right) \left( \sum_{\substack{\|D_2\|=i-k+1 \\ b(D_2)=j-l+1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right)$$

The proof relies crucially on the Proposition 5.7, which tells us in how many ways two given trees can be grafted together.

If we decompose trees or chord diagrams by their left and right subtree, this is a well defined operation. However, if we start with two trees it is not clear which labeling the diamond operation should give and Proposition 5.7 tells us what labelings are possible for it. This being said, we define the diamond operation in the following way

**Definition 6.1** (Diamond operation on trees and chord diagrams). *Let  $T_1, T_2 \in T(\mathcal{R})$ ,  $\lambda$  be a leaf labeling of size  $l(T_1) + l(T_2)$  where  $l(\cdot)$  denotes the numbers of leaves, then we define  $T_1 \diamond_\lambda T_2$  to be the unique tree that has  $T_1$  as left tree,  $T_2$  as right tree and  $\lambda$  as leaf labeling. If we take the induced labeling of a tree  $T$ , we write  $T_1 \diamond_T T_2$ . Analogously, we write for chord diagrams  $D_1, D_2$  and a chord labeling  $\mu$  of size  $|D_1| + |D_2|$   $D_1 \diamond_\mu D_2$  and for the induced labeling of a chord diagram  $C$ , we write  $D_1 \diamond_C D_2$ . In the case no labeling is assigned, i.e.  $T_1 \diamond T_2$  resp.  $D_1 \diamond D_2$  is defined to be the set of all possible labelings.*

**Example 6.2.** Consider  $C =$    $\circ_k$    $(3)$  Then depending on  $k$  we have the

following  $C$  and  $C = D_1 \diamond_C D_2$



k	1	2	3
C			
$T = H_1 \diamond_T H_2$			
$H_1$			
$H_2$			
$D_1$			
$D_2$			

The interplay between the diamond operation and the root share decomposition will be an important tool for proofs. The form of their relationship depends on the insertion place of the root share decomposition as stated below:

**Proposition 6.3.** *Let  $C$  be a chord diagram with  $|C| \geq 3$  and  $C = C' \circ_k C_2$  where  $C_2 = C''' \diamond_{C_2} C''''$ , then*

$$C' \circ_k (C'' \diamond_{C_2} C''') = \begin{cases} (C' \circ_{k-1} C''') \diamond_C C'''' & |C''| \leq k-1, k > 1 \\ C' \diamond_C (C'' \diamond_{C_2} C''') & k = 1 \\ C'' \diamond_C (C' \circ_{k-|C''|-1} C''') & \text{else} \end{cases}$$

*Proof.* Let  $|C| \geq 3$  with  $C = C' \circ_k C_2$ . There are three cases to consider if we look at  $T = T(C)$ :

- (1)  $k$  is the root of  $T$ : This is the case  $k = 1$  and so root share decomposition and diamond decomposition coincide.
- (2)  $k$  lies in the left subtree of  $T$ : Let  $T(D_1)$  be the left subtree of  $T$  and  $D'_1$  the diagram corresponding chord diagram to left subtree of  $T(D_1)$ , then

$$D_1 = C' \circ_{k-1} D'_1$$

- (3)  $k$  lies in the right subtree of  $T$ : Let  $T(D_2)$  be the right subtree of  $T$  and  $D'_2$  the diagram corresponding chord diagram to left subtree of  $T(D_2)$ , then

$$D_2 = C' \circ_{k-1-|D_1|} D'_2$$

□

We have some control over the base chord under the diamond decomposition as seen in the following lemma which is Lemma 4.1 of [7].

**Lemma 6.4** (Triangle inequality for the base chords).

$$b(D_1 \diamond D_2) \leq b(D_1) + b(D_2)$$

Again we see the interplay between the strengths and weaknesses of  $\circ$  and  $\diamond$ . The terminal sets are under control for  $\circ_k$  but we don't know what they do for  $\diamond$ . The branch-left vectors are under control for  $\diamond$  (let  $C = D_1 \diamond D_2$ , then the only component that is increased is the base chord of  $D_2$ ) but we know only partial results on  $\circ_k$ . To prove equation 4 we need the following Lemma which explains how the weighted monomials of chord diagrams behave under the diamond operation.

**Lemma 6.5.** *Let  $C \in \mathcal{R}$  with  $|C| \geq 2$  and  $T, H_1, H_2, D_1, D_2$  as before. Let  $d = d_{b(D_1)}$  be the decoration of the smallest terminal chord of  $D_1$ , then*

$$\omega_{\hat{C}} \hat{a}_C = \omega_{D_1} \omega_{\hat{D}_2} \hat{a}_{D_1} \hat{a}_{D_2} a_{d, b(D_1) + b(D_2) - b(C)}$$

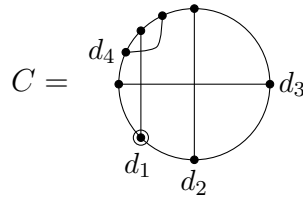
*Proof.* The Lemma follows from the following two claims:

Claim 1:  $\omega_{\hat{C}} = \omega_{D_1} \omega_{\hat{D}_2}$ . Remember that  $\omega_{\hat{C}} = \prod_{k \neq b(C)} \binom{sd_k + \nu_k - 2}{\nu_k}$  and notice that in the induced labeling the base chord of  $D_2$  and  $C$  are the same, so we have  $b(D_2) = b(C)$  because it is the fully right branch leaf of the tree that correspond to  $D_2$  as well that of  $C$ , hence the product of chords is the same on both sides.

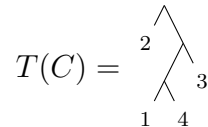
Claim 2:  $\hat{a}_C = \hat{a}_{D_1} \hat{a}_{D_2} a_{d, b(D_1) + b(D_2) - b(C)}$ . This is Lemma 4.11 of [7].

□

**Example 6.6.** Consider the following chord diagram with arbitrary decorations  $d_1, \dots, d_4$  and arbitrary  $s \neq 1$ :



It has terminals  $\text{ter}(C) = \{3, 4\}$ , so  $b(C) = 3$ . The corresponding tree is



so all in all we have for the left hand side of the previous lemma:

$$\omega_{\hat{C}} \hat{a}_C = (d_4 s - 1) a_{d_4, 1} a_{d_1, 0} a_{d_2, 0}$$

For the right hand side we have the following trees and diagrams (the decoration is inherited but the labeling is normalized):

$$D_1 = \begin{array}{c} \circ \\ | \\ \bullet \\ 1, d_2 \end{array} \quad D_2 = \begin{array}{c} 3, d_4 \\ \circ \\ | \\ \circ \\ | \\ \bullet \\ 1, d_1 \end{array} \quad H_1 = \begin{array}{c} | \\ 1 \end{array} \quad H_2 = \begin{array}{c} \wedge \\ 1 \quad 3 \end{array} \quad 2$$

So we have for the right hand side:

$$\begin{aligned} \hat{a}_{D_1} &= 1 \\ \hat{a}_{D_2} &= a_{d_4,1} a_{d_1,0} \\ \omega_{D_1} &= 1 \\ \omega_{\hat{D}_2} &= d_4 s - 1 \\ a_{d(b(D_1)), b(D_1)+b(D_2)-b(C)} &= a_{d_2,0} \end{aligned}$$

multiplying this we indeed get the same as the left hand side as stated in Lemma 6.5.

**Lemma 6.7.** For  $j \in \mathbb{Z}_{\geq 0}$  and every  $k \in \mathbb{N}$

$$\sum_{\substack{\|C\|=i+1 \\ d_{j+1}=1 \\ \nu_{j+1}=n \\ b(C)=j+1}} \hat{a}_C \omega_{\hat{C}} = \sum_{\substack{\|C\|=i+k \\ d_{j+1}=k \\ \nu_{j+1}=n \\ b(C)=j+1}} \hat{a}_C \omega_{\hat{C}}$$

*Proof.* Let

$$\mathcal{C}_{i,j,k} := \{C \in \mathcal{R}^{\text{dec}} : \|C\| = i+k, d_{j+1} = k, \nu_{j+1} = n, b(C) = j+1\}$$

We need to show that

$$A_{i,j} := \{\hat{a}_C \omega_{\hat{C}} : C \in \mathcal{C}_{i,j,1}\} \text{ is in bijection with every } A_{i,j,k} := \{\hat{a}_C \omega_{\hat{C}} : C \in \mathcal{C}_{i,j,k}\}$$

For a fixed  $k$  the map that replaces the decoration  $d_{j+1} = 1$  by  $k$  clearly defines a bijection between  $\mathcal{C}_{i,j,1}$  and  $\mathcal{C}_{i,j,k}$ . This map lifts to a bijection  $A_{i,j} \leftrightarrow A_{i,j,k}$  because the decoration  $d_{j+1}$  is ignored by definition of  $\hat{a}_C$  and  $\omega_{\hat{C}}$ , since  $b(C) = j+1$ .  $\square$

**Proposition 6.8** (Decorated version of Proposition 4.3 from [7]).

$$\sum_{\substack{\|C\|=i+1 \\ b(C)=j+1}} \omega_{\hat{C}} \hat{a}_C = \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} \left( \sum_{\substack{\|D_1\|=k \\ b(D_1) \geq l}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-l} \right) \left( \sum_{\substack{\|D_2\|=i-k+1 \\ b(D_2)=j-l+1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right)$$

*Proof.* We know that each chord diagram  $C$  of size  $i+1$  can be tree decomposed to  $C = D_1 \diamond_C D_2$  and that in this case  $b(D_1) + b(D_2) \geq b(C)$  by the triangle inequality 6.4. However, given  $b(D_1) \geq l$  for fixed  $l$  and  $b(D_2) = j-l+1$  for fixed  $j$  there are  $\binom{j}{l}$  possibilities for  $D_1, D_2$  such that  $C = D_1 \diamond_C D_2$  by Proposition 5.7. Furthermore in all cases  $b(C) = j+1$ . Therefore the sum on the left hand side of the statement splits as follows:

$$\sum_{\substack{\|C\|=i+1 \\ b(C)=j+1}} = \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} \left( \sum_{\substack{\|D_1\|=k \\ b(D_1) \geq l}} \right) \left( \sum_{\substack{\|D_2\|=i-k+1 \\ b(D_2)=j-l+1}} \right)$$

Now given a monomial  $\omega_{\hat{C}} \hat{a}_C$  we know how to decompose it into the monomials as needed by Lemma 6.5. Inserting them into the sums proves the proposition.  $\square$

**Example 6.9.** Let  $i = 3, j = 1$  and  $N \geq 3$  and  $s \in \mathbb{Z}_{\geq 2}$  be arbitrary. We have to consider all chord diagrams with  $\|C\| = 4, b(C) = 2$ . For the decorations we need to consider all compositions of four:

$$(1, 1, 1, 1), (2, 1, 1), (1, 2, 1), (1, 1, 2), (2, 2), (1, 3), (3, 1)$$

Because of the constraint  $b(C) = 2$  all the chord diagrams we need to consider are

$$\mathcal{R}_{\{2,3,4\}}, \mathcal{R}_{\{2,4\}}, \mathcal{R}_{\{2,3\}}, \mathcal{R}_{\{2\}}$$

For  $\mathcal{R}_{\{2,3,4\}}$  and  $\mathcal{R}_{\{2,4\}}$  only the decoration  $d_1 = d_2 = d_3 = d_4 = 1$  is possible so we calculate these: We have one chord diagram with  $ter = \{2, 3, 4\}$  and branch left vector  $(0, 1, 1, 1)$ :

$$\hat{a}_C = a_1^2 a_0 \text{ and } \omega_{\hat{C}} = (s - 1)^2$$

There are three chord diagrams with  $ter = \{2, 4\}$ , namely the two chord diagrams with branch left vector  $(0, 1, 0, 2)$ :

$$\hat{a}_C = a_2 a_0^2 \text{ and } \omega_{\hat{C}} = \binom{s}{2}$$

and the one with branch left vector  $(0, 1, 1, 1)$ :

$$\hat{a}_C = a_2 a_0^2 \text{ and } \omega_{\hat{C}} = (s - 1)^2$$

Summing this up  $\mathcal{R}_{\{2,3,4\}}$  and  $\mathcal{R}_{\{2,4\}}$  contribute to the left hand side by:

$$(s - 1)^2 a_1^2 a_0 + a_2 a_0^2 \left( 2 \binom{s}{2} + (s - 1)^2 \right) = (s - 1)^2 a_1^2 a_0 + (2s^2 - 3s + 1) a_2 a_0^2$$

$\mathcal{R}_{\{2,3\}}$  consists only of one chord diagram with branch left vector  $(0, 1, 1)$  and we have to consider the decorations:  $(d_1, d_2, d_3) \in \{(2, 1, 1), (1, 2, 1), (1, 1, 2)\}$ , so it contributes on the left hand side with:

$$((2s - 1) + 2(s - 1)) a_0 a_1 = (4s - 3) a_0 a_1$$

$\mathcal{R}_{\{2\}}$  has only one chord diagram which contributes to left hand side by  $3a_0$ . The weight is 1 because the branch left vector of the chord diagram is  $(0, 1)$  where the second coordinate is ignored by  $\omega_{\hat{C}}$ , but there are three chord diagrams to consider, namely those that are decorated by  $(1, 3), (3, 1)$  and  $(2, 2)$ . So the left hand side of previous lemma is for this example:

$$(s - 1)^2 a_1^2 a_0 + (2s^2 - 3s + 1) a_2 a_0^2 + (4s - 3) a_0 a_1 + 3a_0$$

For the right hand side we need to take the induced labels of  $D_1$  and  $D_2$  and we have to consider only the last sum:

$$\begin{aligned} \text{RHS} &= \left[ \sum_{\substack{\|D_1\|=1 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} \right] \cdot \left[ \sum_{\substack{\|D_2\|=3 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right] \\ &+ \left[ \sum_{\substack{\|D_1\|=2 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} \right] \cdot \left[ \sum_{\substack{\|D_2\|=2 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right] \\ &+ \left[ \sum_{\substack{\|D_1\|=3 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} \right] \cdot \left[ \sum_{\substack{\|D_2\|=1 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right] \end{aligned}$$

Since  $N \in \mathbb{N}_{\geq 3}$ , we have

$$\sum_{\substack{\|D_2\|=3 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} = \sum_{\substack{\|D_2\|=2 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} = \sum_{\substack{\|D_2\|=1 \\ b(D_2)=1}} \omega_{\hat{D}_2} \hat{a}_{D_2} = 1$$

$$\sum_{\substack{\|D_1\|=1 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} = a_0$$

$$\sum_{\substack{\|D_1\|=2 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} = (s-1)a_0a_1 + a_0$$

$$\sum_{\substack{\|D_1\|=3 \\ b(D_1) \geq 1}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-1} = (s-1)^2a_0a_1^2 + (2s^3 - 3s + 1)a_0^2a_2 + ((s-1) + (2s-1))a_0a_1 + a_0$$

Summing this up, we indeed get the left hand side.

**Proposition 6.10** (Restricted decorated version of Proposition 4.3 of [7]).

$$\sum_{\substack{\|C\|=i+1 \\ b(C)=j+1 \\ \nu_{b(C)}=n}} \omega_{\hat{C}} \hat{a}_C = \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} \left( \sum_{\substack{\|D_1\|=k \\ b(D_1) \geq l}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-l} \right) \left( \sum_{\substack{\|D_2\|=i-k+1 \\ b(D_2)=j-l+1 \\ \nu_{b(D_2)}=n-1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right)$$

*Proof.* Note that  $\nu_{b(D_1 \diamond D_2)}(D_1 \diamond D_2) = \nu_{b(D_2)}(D_2) + 1$ . Indeed,  $b(D_1 \diamond D_2) = b(D_2)$  and attaching a tree on the left side of  $T(D_2)$  increments the rightmost branch, which ends at the leaf  $b(D_2)$ . Therefore the length of the rightmost branch matches as given in the proposition. The rest follows by Proposition 6.8  $\square$

The key idea in generalizing the results of this section from the results of [7] was to break up the sums in the lemmas above according to the branch left value of the base chord. This further suggests that the branch left vector and the rooted trees are not just technical tools, but are showing something important about the structure of chord diagrams.

## 7. BRIDGE EQUATION AND MAIN RESULT

In this section we develop an equation that builds a bridge between the sum that contains only chord diagrams and the sum that is calculated by derivations which will finally let us connect  $G^{\text{dif}}$  and  $G^{\text{comb}}$  (see Section 4 for the definitions of  $G^{\text{dif}}$  and  $G^{\text{comb}}$ ). Therefore, we will call it the bridge equation. It is proved by induction. The following lemma gives the base case whereas the lemma after it states the general case.

**Lemma 7.1.** *Let  $i \geq 1, j \geq 1$ , then*

$$\sum_{\substack{\|C\|=i+1 \\ d_{j+1}=1 \\ \nu_{j+1}=1 \\ b(C)=j+1}} \hat{a}_C \omega_{\hat{C}} = [x^i] \left( \sum_{l \geq 1} \frac{g_l^{\text{comb}}(x)}{l!} \partial_{\rho=0}^l \right) \rho^j$$

*Proof.* Because the right hand side of the equation is only non-zero when  $l = j$ , it suffices to prove

$$(5) \quad \sum_{\substack{\|C\|=i+1 \\ d_{j+1}=1 \\ \nu_{j+1}=1 \\ b(C)=j+1}} \hat{a}_C \omega_{\hat{C}} = \sum_{\substack{\|D\|=i \\ b(D) \geq j}} \omega_D \hat{a}_D a_{d(b(D)), b(D)-j}$$

Consider first  $D$  satisfying the conditions on the right hand side of (5). Let  $C = D \diamond D_2$  with  $D_2$  being the chord diagram with only one chord. We have the triangle inequality for the label of the base chords:  $b(D) \geq b(C) - b(D_2)$ . Note that there indeed exists only one  $C$  with this diamond decomposition and  $b(C) = j + 1$ ; this is because there is only one compatible shuffle which ends with the integer  $j + 1$ . Now, in the sum of the left hand side of (5) every chord diagram  $C$  splits into  $D \diamond D_2$  with  $D_2$  being the chord diagram with only one chord since  $\nu_{j+1} = 1$ . Thus  $D$  now has size  $\|D\| = \|C\| - d_{j+1} = i$  and its base chord satisfies  $b(D) \geq j + 1 - 1 = j$ . Thus the chord diagrams on each side of (5) correspond via  $C = D \diamond D_2$  and it is left to see that the corresponding summands are actually equal. This was already done in Lemma 6.5.  $\square$

Now, we can use this result to prove the statement that we are looking for:

**Lemma 7.2** (Bridge equation). *Let  $n \geq 1$ , then*

$$\sum_{\substack{\|C\|=i+1 \\ d_{j+1}=1 \\ \nu_{j+1}=n \\ b(C)=j+1}} \hat{a}_C \omega_{\hat{C}} = [x^i] \left( \sum_{l \geq 1} \frac{g_l^{\text{comb}}(x)}{l!} \partial_{\rho=0}^l \right)^n \rho^j$$

*Proof.* For better readability define:  $G_{\partial\rho}(x) := \sum_{l \geq 1} \frac{g_l^{\text{comb}}(x)}{l!} \partial_{\rho=0}^l$  and  $F_{i,j,n} := [x^i] G_{\partial\rho}^n(x) \rho^j$ . Let  $i, j$  be fixed; we prove the statement by induction over  $n$ . For  $n = 1$  the statement is

true by the previous Lemma. Now analogously to the proof of Lemma 4.14 in [7], we observe that:

$$\begin{aligned}
F_{i,j,n} &= [x^i]G_{\partial\rho}^n(x)\rho^j \\
&= \sum_{k=1}^i ([x^k]G_{\partial\rho}(x)) ([x^{i-k}]G_{\partial\rho}^{n-1}(x)) \rho^j \\
&\stackrel{\text{Leibniz-Rule}}{=} \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} ([x^k]G_{\partial\rho}(x)\rho^l) ([x^{i-k}]G_{\partial\rho}^{n-1}\rho^{j-l}) \\
&\stackrel{\text{Definition}}{=} \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} [x^k]g_l^{\text{comb}}(x) \cdot F_{i-k,j-l,n-1} \\
&\stackrel{\text{induction}}{=} \sum_{k=1}^i \sum_{l=1}^j \binom{j}{l} \left( \sum_{\substack{\|D_1\|=k \\ b(D_1)\geq l}} \omega_{D_1} \hat{a}_{D_1} a_{b(D_1)-l} \right) \left( \sum_{\substack{\|D_2\|=i-k+1 \\ b(D_2)=j-l+1 \\ \nu_{b(D_2)}=n-1}} \omega_{\hat{D}_2} \hat{a}_{D_2} \right) \\
&\stackrel{\text{Prop 6.10}}{=} \sum_{\substack{\|C\|=i+1 \\ d_{j+1}=1 \\ \nu_{j+1}=n}} \hat{a}_C \omega_{\hat{C}}
\end{aligned}$$

□

The next result tells us that  $g_1^{\text{comb}}$  satisfies the same recurrence as given by the Dyson-Schwinger equation itself for  $g_1^{\text{dif}}$ .

**Lemma 7.3.** *Let  $G_{\partial\rho}^{\text{comb}} := \sum_{l\geq 1} \frac{g_l^{\text{comb}}}{l!}$  and  $\tilde{F}_k(\rho) := \sum_{l\geq 0} a_{k,l}\rho^l$ , then*

$$g_1^{\text{comb}} = \sum_{k=1}^N x^k \sum_{n\geq 0} \binom{n+sk-2}{n} (G_{\partial\rho}^{\text{comb}})^n \tilde{F}_k(\rho)$$

*Proof.* Consider the coefficient of  $x^i$  on the right hand side of the equation that we want to show.

$$[x^i]\text{RHS} = \sum_{k=1}^N \sum_{n\geq 0} \binom{n+sk-2}{n} \sum_{l\geq 0} a_{k,l} [x^{i-k}] (G_{\partial\rho}^{\text{comb}})^n \rho^l$$

Now, we can use the bridge equation and Lemma 6.7:

$$\sum_{\substack{\|C\|=i+k \\ d_{l+1}=k \\ \nu_{l+1}=n \\ b(C)=l+1}} \hat{a}_C \omega_{\hat{C}} = [x^i] (G_{\partial\rho}^{\text{comb}})^n \rho^l \quad \text{for all } k = 1 \dots N.$$

So we get:

$$\begin{aligned}
[x^i]\text{RHS} &= \sum_{k=1}^N \sum_{n \geq 0} \binom{n + sk - 2}{n} \sum_{l \geq 0} a_{k,l} \sum_{\substack{\|C\|=i \\ d_{l+1}=k \\ \nu_{l+1}=n \\ b(C)=l+1}} \hat{a}_C \omega_{\hat{C}} \\
&= \sum_{k=1}^N \sum_{n \geq 0} \sum_{l \geq 0} a_{k,l} \sum_{\substack{\|C\|=i \\ d_{l+1}=k \\ \nu_{l+1}=n \\ b(C)=l+1}} \hat{a}_C \omega_{\hat{C}} \binom{\nu_{l+1} + sk - 2}{\nu_{l+1}} \\
&= \sum_{k=1}^N \sum_{n \geq 0} \sum_{l \geq 0} \sum_{\substack{\|C\|=i \\ d_{b(C)}=k \\ \nu_{b(C)}=n \\ b(C)=l+1}} a_{k,b(C)-1} \hat{a}_C \omega_{\hat{C}} \binom{\nu_{l+1} + sk - 2}{\nu_{l+1}}
\end{aligned}$$

Since we have  $l = b(C) + 1$ , we see that

$$\omega_{\hat{C}} \binom{\nu_{l+1} + sk - 2}{\nu_{l+1}} = \omega_C$$

Look carefully at the restrictions of the last sum. We need to verify that we can drop the last three constraints because we are summing over all possible  $k, n, l$ :

- (1) The restriction  $d_{b(C)} = k$  drops because we are summing over all  $k = 1 \dots N$ .
- (2) The restriction of  $\mathcal{R}_{dec}$  to  $\|C\| = i, \nu_{b(C)} = n$  is always non-empty for some  $n$  and summing over all  $n$  indeed yields all rooted connected decorated chord diagrams with  $\|C\| = i$ .
- (3)  $b(C) \geq 1$  so we can drop  $b(C) = l + 1$  and the sum over  $l$ .

In conclusion

$$[x^i]\text{RHS} = \sum_{\|C\|=i} \omega_C \hat{a}_C a_{d(b(C)), b(C)-1} = g_1^{\text{comb}}$$

□

**Theorem 7.4.** *The analytic Dyson-Schwinger equation*

$$G(x, L) = 1 - \sum_{k \geq 1} x^k G(x, \partial_{-\rho=0})^{1-sk} (e^{-L\rho} - 1) F_k(\rho) \text{ where } F_k(\rho) = \sum_{l \geq 0} a_{k,l} \rho^{l-1}$$

has as formal solution the following combinatorial expansion in terms of chord diagrams:

$$G(x, L) = 1 - \sum_{k \geq 1} \frac{(-L)^k}{k!} \sum_{b(C) \geq k} \omega_C \hat{a}_C a_{d_b(C), b(C)-k} x^{\|C\|}$$

*Proof.* Since the renormalization group equation is true for both  $g^{\text{comb}}$  (by Theorem 4.7) and  $g^{\text{dif}}$  (by Proposition 2.1),  $G^{\text{dif/comb}}$  are built from  $g_1^{\text{dif/comb}}$  in completely the same way:

$$g_k^{\text{dif/comb}} = g_1^{\text{dif/comb}} (sx\partial_x - 1) g_{k-1}^{\text{dif/comb}} \text{ for } k \geq 2$$



For  $g_1^{\text{dif}}$ , by applying the generalized geometric series to  $G^{\text{dif}}(x, \partial_{-\rho})^{1-sk}$  in the Dyson-Schwinger equation we have:

$$g_1^{\text{dif}} = \sum_{k=1}^{\infty} x^k \sum_{n \geq 0} \binom{n + sk - 2}{n} (G_{\partial\rho}^{\text{dif}})^n \tilde{F}_k(\rho)$$

where  $\tilde{F}_k(\rho) = (e^{-L\rho} - 1)^{\frac{1}{\rho}} \sum_{l \geq 0} a_{k,l} \rho^l$ . So, by Lemma 7.3 for  $g_1^{\text{comb}}$  and the Dyson-Schwinger equation itself for  $g_1^{\text{dif}}$

$$g_1^{\text{comb}}(x) - g_1^{\text{dif}}(x) = \sum_{k=1}^N x^k \sum_{n \geq 0} \binom{n + sk - 2}{n} ((G_{\partial\rho}^{\text{comb}})^n - (G_{\partial\rho}^{\text{dif}})^n) \tilde{F}_k(\rho)$$

Write  $g_1^{\text{comb}} = \sum_{k \geq 1} g_{1,k}^{\text{comb}} x^k$  and similarly for  $g_1^{\text{dif}}$ . The first coefficients of  $g_1^{\text{comb}}$  and  $g_1^{\text{dif}}$  are the same. Apply the renormalization group equation to convert the bigger  $g_k$  to  $g_1$ ; note that the coefficient of  $x^k$  in the equation above only involves  $g_{1,\ell}$  with  $\ell < k$  on the right hand side. So inductively we obtain  $g_{1,k}^{\text{comb}} = g_{1,k}^{\text{dif}}$  and so

$$g_1^{\text{comb}}(x) - g_1^{\text{dif}}(x) = 0$$

□

## 8. CONCLUSION

What we have achieved in our main theorem, Theorem 7.4 is to solve a family of Dyson-Schwinger equations as expansions over decorated rooted connected chord diagrams. The analytic input to the expansion is the expansion of the regularized integrals for the primitive graphs building the Dyson-Schwinger equation. Each decorated chord diagram in the expansion contributes a weighted monomial in the coefficients of the expansions of the primitives.

Compared to [7] the achievement here has been the generalization to a substantially larger class of Dyson-Schwinger equations. This indicates that these chord diagram expansions are not mere curiosities but actually quite general. Whether or not this would be the case was one of the main open questions after [7]. This generalization also clarifies some of the technical underpinnings, in particular showing the key role that the binary rooted tree associated to a chord diagram plays.

Furthermore, in the course of these investigations one of us (MH) wrote code to calculate chord diagrams, their weights and all the associated objects which we define. This code is available in [6].

There are a few obvious things to consider next. First would be a generalization to systems of Dyson-Schwinger equations which would capture more of the Dyson-Schwinger equations of interest in physics. Second is to consider the asymptotic consequences of these results. There is a reasonably good understanding of the asymptotics of chord diagrams, but not of the particular parameters important here. This is being investigated in the simpler case of [7] by the second author with Julien Courtiel. Third there are unresolved combinatorial issues. For example the rooted tree construction is not very natural combinatorially, but its importance suggests that it should be. Therefore, there ought to be a more transparent reformulation of the rooted trees. There are also many patterns in the coefficients of our chord diagram expansion which have barely been investigated. See [6] for more details of some of these patterns.

## REFERENCES

- [1] D.J. Broadhurst and D. Kreimer. Exact solutions of Dyson-Schwinger equations for iterated one-loop integrals and propagator-coupling duality. *Nucl. Phys. B*, 600:403–422, 2001. arXiv:hep-th/0012146.
- [2] Alain Connes and Dirk Kreimer. Hopf algebras, renormalization and noncommutative geometry. *Commun. Math. Phys.*, 199:203–242, 1998. arXiv:hep-th/9808042.
- [3] Kurusch Ebrahimi-Fard and Dirk Kreimer. Hopf algebra approach to Feynman diagram calculations. *J. Phys. A*, 38:R285–R406, 2005. arXiv:hep-th/0510202.
- [4] Loïc Foissy. Faà di Bruno subalgebras of the Hopf algebra of planar trees from combinatorial Dyson-Schwinger equations. *Advances in Mathematics*, 218(1):136–162, 2007. arXiv:0707.1204.
- [5] Loïc Foissy. General Dyson-Schwinger equations and systems. *Communications in Mathematical Physics*, 327(1):151–179, 2014. arXiv:1112.2606.
- [6] Markus Hihn. *The generalized chord diagram expansion*. PhD thesis, Humboldt University of Berlin, (submitted 2015).
- [7] Nicolas Marie and Karen Yeats. A chord diagram expansion coming from some Dyson-Schwinger equations. *Communications in Number Theory and Physics*, 7(2):251–291, 2013. arXiv:1210.5457.
- [8] Walter D. van Suijlekom. Renormalization of gauge fields: A Hopf algebra approach. *Commun. Math. Phys.*, 276:773–798, 2007. arXiv:hep-th/0610137.
- [9] Walter D. van Suijlekom. Renormalization of gauge fields using Hopf algebras. In B. Fauser, J. Tolksdorf, and E. Zeidler, editors, *Recent Developments in Quantum Field Theory*. Birkhauser Verlag, 2008. arXiv:0801.3170v1.
- [10] Karen Yeats. Rearranging Dyson-Schwinger equations. *Mem. Amer. Math. Soc.*, 211, 2011.
- [11] Karen Amanda Yeats. *Growth estimates for Dyson-Schwinger equations*. PhD thesis, Boston University, 2008.