Enumeration of generalized BCI lambda-terms

Olivier Bodini*
Institut Galilée
Univ. Paris 13
Villetaneuse, France
olivier.bodini@lipn.univ-paris13.fr

Danièle Gardy†
PRiSM
Université de Versailles Saint-Quentin
78035 Versailles, France
daniele.gardy@uvsq.fr

Bernhard Gittenberger‡
Institute for Discrete Mathematics and Geometry
Vienna University of Technology
A-1040 Wien, Austria
gittenberger@dmg.tuwien.ac.at

Alice Jacquot*
Institut Galilée
Univ. Paris 13
Villetaneuse, France
alice.jacquot@lipn.univ-paris13.fr

Submitted: Jan 11, 2013; Accepted: Dec 11, 2013; Published: Dec 17, 2013
Mathematics Subject Classifications: 05A16, 05A99, 05C20, 03B40

Abstract

We investigate the asymptotic number of elements of size n in a particular class of closed lambda-terms (so-called $BCI(p)$-terms) which are generalizations of lambda-terms related to axiom systems of combinatory logic. By deriving a differential equation for the generating function of the counting sequence we obtain a recurrence relation which can be solved asymptotically. We derive differential equations for the generating functions of the counting sequences of other more general classes of terms as well: the class of $BCK(p)$-terms and that of closed lambda-terms. Using elementary arguments we obtain upper and lower estimates for the number of closed lambda-terms of size n. Moreover, a recurrence relation is derived which allows an efficient computation of the counting sequence. $BCK(p)$-terms are discussed briefly.

*Supported by ANR Magnum project (France)

†Supported by ANR Boole project (France). This author's work was partially carried out during her sabbatical leave at the Institute for Discrete Mathematics and Geometry, TU Wien, Austria.

‡Supported by FWF grant SFB F50-03 and ÖAD, grant F04/2012.

1 Introduction

Lambda-terms play a prominent role in the theory of computer programming. In order to investigate properties of randomly generated lambda-terms we have to know how many terms of a given size there are. This paper is devoted to the asymptotic counting of particular classes of lambda-terms.

Lambda-terms were invented by Church and Kleene in the 30ies (see [5, 18, 19]) together with a set of rules for manipulating them, the so-called lambda-calculus. This is a very powerful formal language which can be used to describe computer programs, analyze programming languages or investigate decision problems. Moreover, it is the basis of the programming language LISP.

A lambda-term is a formal expression built of variables and a quantifier λ which in general occurs more than once and acts on one of the free variables. It can be described by the context-free grammar $T ::= a \mid (T * T) \mid \lambda a.T$ where a is a variable. The concatenation of terms is called *application* and adding the prefix λa to a term is called *abstraction*. Each abstraction *binds* a variable in the whole term following it and each variable can only be bound by at most one abstraction. A term where all the variables are bound is called a *closed* lambda-term, otherwise an *open* lambda-term. For example, $(\lambda x.(x * x) * \lambda y.y)$ is a closed lambda-term whereas $(\lambda x.(x * z) * \lambda y.y)$ is an open one.

Our aim is to study the asymptotic number of closed lambda-terms of a given size when the size is tending to infinity. We define the size of a lambda-term recursively by

$$|x| = 1, \quad |\lambda x.T| = 1 + |T|, \quad |(S * T)| = 1 + |S| + |T|. \tag{1}$$

Moreover, note that we will count lambda-terms up to isomorphism: Only the structure of the bindings is important whereas variable names are unimportant. For closed lambda-terms this is precisely α-conversion (see [1, Ch. 2]). For instance, the terms $\lambda y.(\lambda x.x * \lambda z.y)$, $\lambda y.(\lambda x.x * \lambda x.y)$, $\lambda x.(\lambda y.y * \lambda z.x)$ are considered to be identical. Observe that the second term is obtained from the first one by replacing z by x, which is "by coincidence" the same variable as that in the sub-term $\lambda x.x$ just left to it. But as stated above the important issue is that the last quantifier does not bind the variable following it; therefore the name must only be different from y.

Since the determination of the asymptotic number of lambda-terms seems to be a hard problem (*cf.* the discussion of this issue in [3] and for a similar problem in [13, end of Sec. 3]) we confine ourselves with the asymptotic analysis of a simpler subclass of lambda-terms and give an outlook to the analysis of a larger and more complicated subclass. The classes considered are $BCI(p)$- and $BCK(p)$-terms and unless explicitly stated we mean closed terms. The names stem from the correspondence of $BCI(1)$- and $BCK(1)$-terms to the logical systems BCI and BCK, respectively, which are studied in combinatory logic (see [16, 15, 17]). $BCI(1)$-terms are also known as linear lambda-terms, $BCK(1)$-terms as affine lambda-terms. Due to the Curry-Howard isomorphism [8, 20] $BCI(1)$-terms constitute proofs of intuitionistic tautologies in which every propositional variable occurs exactly twice. One might be tempted to think that an analogous statement holds for $p > 1$. However, this is false since such terms have to be typable and already for $p = 2$ we easily find non-typable $BCI(2)$-terms, e.g. $(\lambda x.x * x) * (\lambda x.x * x)$.

The plan of the paper is as follows: In the next section we state our notations, definitions and some immediate observations. In Section 3 we derive the functional equations for the generating functions corresponding to $BCI(p)$-terms, $BCK(p)$-terms as well as general closed lambda-terms. Then we derive the asymptotic order of the number of $BCI(p)$-terms (Section 4). Section 5 is devoted to an upper and a lower estimate for the number λ_n of closed lambda-terms of size n. This is done using rather elementary arguments, but it is still sufficient to obtain the asymptotic main term of $\log \lambda_n$. Moreover, we derive a recurrence relation which allows an efficient computation of the numbers λ_n. In the final section, we briefly discuss $BCK(p)$-terms.

The enumeration of $BCI(1)$-terms was carried out by Bodini et al. [2] by constructing a nice bijection to certain diagrams. They showed that the number of $BCI(1)$-terms of size n is asymptotically

$$\frac{C}{n^{1/6}}\left(\frac{2n}{e}\right)^{n/3} \tag{2}$$

if $n \equiv 2 \bmod 3$ and zero otherwise. They obtained also the asymptotic number of $BCK(1)$-terms which differs from (2) by a multiplicative factor $e^{\frac{1}{2}(2n)^{2/3}-\frac{1}{6}(2n)^{1/3}}$. A quantitative comparison of provable formulas between BCI(1) and BCK(1) was done in [12].

Models with a different notion of size (leaves do not contribute to the size, i.e. they have weight zero) were studied in [7, 13]. In [7] upper and lower bounds for the counting sequence were derived and questions like typability were discussed. The paper [13] approaches the counting problem by representations of terms using de Bruijn indices. They derive recurrence relations for the number of terms with or without constraints on the number of free variables and discuss the issue of random generation of terms as well. This allows an efficient computation and experimental analysis of term properties like typability or some shape characteristics.

2 Notation and basic facts

A lambda-term can be regarded as a so-called *enriched tree* which is a particular directed acyclic graph. In fact, consider a Motzkin tree (i.e., a rooted unary-binary tree) and add directed edges connecting a unary node and a leaf such that each leaf is “bound” by a directed edge from exactly one of the unary nodes that are its ancestors in the tree. The correspondence is obvious (see Figure 1): leaves correspond to variables, unary nodes to abstractions, binary nodes to applications and the additional directed edges to the binding relations between abstractions and variables. Clearly, since all leaves are bound, the lambda-term is closed. Of course, open lambda-terms can be represented in an analogous manner by a directed acyclic graph where some leaves have in-degree zero (that means that they have no ingoing *directed* edge).

We will not distinguish between a lambda-term and its enriched tree representation. In addition, when speaking of lambda-terms, we will utilize the following abuse of the wording: A *unary node* of a lambda-term is a unary node (i.e. node of out-degree one) of the underlying Motzkin tree (i.e. a node becoming unary if all directed edges are

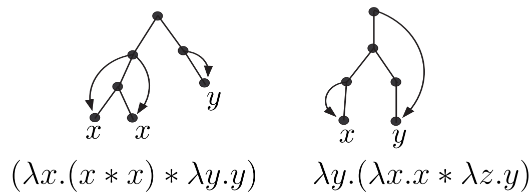

Figure 1: Two enriched trees and the closed lambda-terms corresponding to them. Note that the node labels can be omitted, since $(\lambda x.(x*x)*\lambda y.y)$ and $(\lambda a.(a*a)*\lambda b.b)$ are the same term.

removed). These are precisely the nodes corresponding to abstractions. Analogously, we call the nodes corresponding to applications *binary nodes* and nodes corresponding to variables *leaves* of the lambda-term. In a strict sense, leaves have always degree one and in-degree one as well (i.e. each leaf x is incident with exactly one *undirected* and exactly one *directed* edge pointing towards x).

Moreover, we distinguish between *edges*, i.e. edges of the underlying Motzkin tree, and *pointers*, i.e. directed edges from a unary node to a leaf.

Definition 1.

- $BCI(p)$ is the set of (non-empty) closed lambda-terms where each unary node has *exactly* p pointers, i.e. binds exactly p occurrences of its variable.
- $BCK(p)$ is the set of closed lambda-terms where each unary node binds *at most* p leaves.

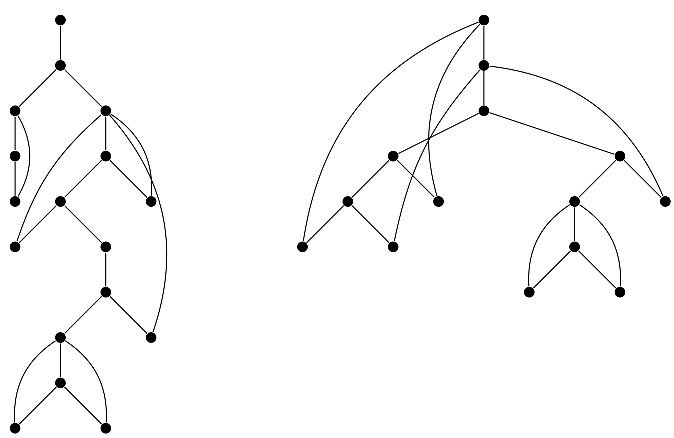

Figure 2: Left: a closed λ-term of size 17. Center: a term in $BCI(2)$ of size 14. Right: a term in $BCK(1)$ of size 15.

A lambda-term from $BCI(p)$ has three types of nodes: unary nodes (which are actually of arity $p+1$, as there are p pointers going from this node to leaves), binary nodes, and leaves. The size of such a lambda-term is the total number of its nodes. We start with some obvious observations:

Fact 2. *The smallest terms of $BCI(p)$ have one unary node at the root and p leaves. There are p pointers from the root to all the leaves. Obviously, if we remove the root and all its pointers, we are left with a binary tree. Thus the number of such terms is equal to the number of binary trees with $p-1$ binary nodes and p leaves. This is precisely the Catalan number $C_{p-1} = \binom{2p-2}{p-1}/p$. And clearly, the size of all these terms is $2p$.*

Fact 3. *A term of $BCI(p)$ with j unary nodes has pj leaves and $pj-1$ binary nodes; its size is therefore equal to $(2p+1)j-1$.*

3 The generating functions for various classes of closed lambda-terms

We will enumerate lambda-terms by means of generating functions. Let $g_n = g_n^{(p)}$ be the number of $BCI(p)$-terms of size n and $G_p(z)$ be the generating function of this sequence. By Fact 2 we have actually

$$G_p(z) = \sum_{j\geqslant 1} g_{j(2p+1)-1} z^{j(2p+1)-1}.$$

Analogously, define $F_p(z) = \sum_{n\geqslant 1} f_n z^n$ and $\Lambda(z) = \sum_{n\geqslant 1} \lambda_n z^n$ where $f_n = f_n^{(p)}$ is the number of $BCK(p)$-terms of size n and λ_n the number of closed lambda-terms of size n.

The next step is the setting up of functional equations for the generating functions. This will be done by giving a formal specification of the combinatorial objects and then using the symbolic method (see [9]). From [2] we already know that $G_1(z)$ satisfies the equation

$$G_1(z) = z^2 + zG_1(z)^2 + \Delta_1 G_1(z),$$

where the differential operator Δ_1 is $2z^4 D$ and D denotes the ordinary differential operator.

Proposition 4. *The generating function of $BCI(p)$-terms satisfies the differential equation*

$$G_p(z) = C_{p-1} z^{2p} + zG_p(z)^2 + \Delta_p G_p(z) \tag{3}$$

where

$$\Delta_p = \sum_{l=1}^{p} \frac{\alpha_{l,p}}{l!} z^{l+2p+1} D^l \tag{4}$$

with constants $\alpha_{l,p}$ defined by

$$\alpha_{l,p} = \sum_{\sum_i s_i = l;\ \sum_i i s_i = p} \binom{l}{s_1, \ldots, s_p} \prod_{m=1}^{p} \binom{2m}{m}^{s_m}. \tag{5}$$

Proof. A $BCI(p)$-term can be specified by the formal equation

$$\mathcal{T} = \mathcal{S} \cup (\{\circ\} \times \mathcal{T} \times \mathcal{T}) \cup (\{\circ\} \times \tilde{\mathcal{T}}), \tag{6}$$

where the set $\mathcal{S}$ is the set of all smallest $BCI(p)$-terms (*cf.* Fact 1) and $\tilde{\mathcal{T}}$ a certain set of open $BCI(p)$-terms.[1] This can be explained as follows: A $BCI(p)$-term falls into exactly one of three categories: It is either

- a smallest term,
- or its root is a binary node and the two sub-terms attached to the root are themselves $BCI(p)$-terms,
- or its root is a unary node and the sub-term attached to the root is an open $BCI(p)$-term with exactly p free leaves.

In order to specify all $BCI(p)$-terms and avoid ambiguities, we have to take some care in the choice of $\tilde{\mathcal{T}}$. Indeed, each $BCI(p)$-term will be generated exactly once by the specification (6) if we generate $\tilde{\mathcal{T}}$ by starting with a $BCI(p)$-term and then generating p leaves and connecting them to the unary root node by a pointer in the following way. To construct a term $\tilde{t} \in \{\circ\} \times \tilde{\mathcal{T}}$, choose a BCI-term t and p nodes of t, where multiple choices of a node are allowed. Each node v corresponds to an edge, namely the edge leading to v if v is not the root and the edge connecting v with the new root (of the term $\tilde{t} \in \{\circ\} \times \tilde{\mathcal{T}}$) otherwise. Thus the choice of the p nodes "hits" edges of the term $\tilde{t}$. Assume that l edges are hit and s_i of them exactly i times.

If an edge is hit i times, then replace it by a path where at each node of the path a binary tree is attached, either to the left or to the right of the path, and the number of leaves of all these binary trees altogether is equal to i (see Figure 3 for an illustration of this process). Thus the replacement creates i new leaves and i new internal nodes. The whole replacement process creates exactly $\sum_{i=1}^{p} is_i = p$ new leaves and p new internal nodes in t. Therefore $\tilde{t}$ has exactly $2p+1$ more nodes than t and obviously $\tilde{t}$ is an open $BCI(p)$-term. Conversely, if we have a $BCI(p)$-term with a unary root node, then removing it together with its pointers yields a term with p free leaves. These leaves must be children of a binary node since otherwise the parent node must have pointers to p descendants which is impossible. Thus the free leaves induce a set of subtrees of t which are binary trees with free leaves only.

Now let us count in how many ways this can be done. Each edge which is hit i times is actually replaced by a sequence of left or right binary trees. The generating function associated to binary trees is $T(u) = \sum_{n\geqslant 1} C_{n-1}u^n = (1-\sqrt{1-4u})/2$. Thus the number of such sequences with exactly i leaves is

$$[u^i]\frac{1}{1-2T(u)} = [u^i]\frac{1}{\sqrt{1-4u}} = \binom{2i}{i}.$$

[1]Note the slight abuse of notation in (6): The last Cartesian product on the right-hand side is not a Cartesian product in a strict sense, but only on the level of the underlying Motzkin trees, since we will add pointers going from the new root to some leaves of $\tilde{\mathcal{T}}$.

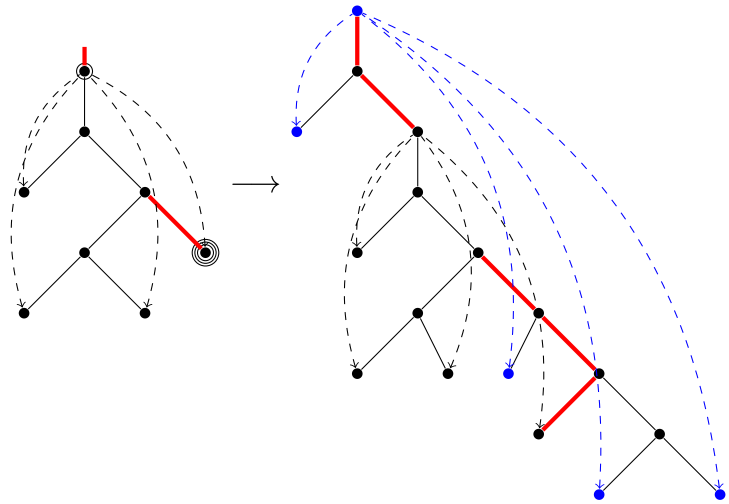

Figure 3: To the left, a BCI(4) term with a node pointed once and another pointed 3 times where pointing at a node is represented by encircling the dot representing it in the figure. So the root on top is pointed at once, the right-most leaf three times. The corresponding hit edges are the thick ones.

At the right, a possible BCI(4) obtained from the left term. Each thick edge has been replaced by a thick path where binary trees have been attached; their leaves are linked to the newly created unary node at the root. The root edge (on top of the left term) has been replaced by a path of length two (having thus three nodes) and a size one tree has been attached to the left at the middle node; the second thick edge of the left term has been replaced by a path of length 3 with two attachments: a size one tree left from the second node and a size 3 tree to the right of the third node of the path.

Note that s_i of the l edges are hit i times, $i = 1, \dots, p$. The number of ways to partition the l edges w.r.t. the multiplicity of the hits is $\binom{l}{s_1,\dots,s_p}$. Then each of the s_i edges which is hit i times is replaced by one of the $\binom{2i}{i}$ possible sequences of binary trees. Therefore there are $\prod_{i=1}^{p} \binom{2i}{i}^{s_i}$ ways of doing the whole replacement. Finally, note that choosing l distinct edges corresponds to applying the operator $z^l D^l / l!$ on the level of generating functions and the $2p+1$ new nodes created during the replacement process yield a factor z^{2p+1}. $\square$

Proposition 5. *Let $F(z)$ denote a formal power series (with real coefficients), $D_u = \partial/\partial u$, the formal derivative, and U the operator $G(u) \mapsto G(0)$, $G(u)$ being a formal power series. Then*

$$\Delta_p F(z) = \frac{z^{2p+1}}{p!} U D_u^p F\left(\frac{z}{\sqrt{1-4u}}\right) = z^{2p+1} [u^p] F\left(\frac{z}{\sqrt{1-4u}}\right).$$

Proof. The second equation is obvious since $U D_u^p / p! = [u^p]$ is exactly Taylor's theorem. For proving the first equation set $D_z = \partial/\partial z$ and $f(u) := 1/\sqrt{1-4u} = \sum_{i \geqslant 0} \binom{2i}{i} u^i$.

Therefore by Faà di Bruno's formula (see e.g. [6, p. 137])[2] we obtain

$$\begin{aligned}\frac{z^{2p+1}}{p!}UD_u^pF(zf(u)) &= \frac{z^{2p+1}}{p!}\sum_{\sum_{i=1}^p is_i=p}\frac{p!}{s_1!\cdots s_p!}(D^{s_1+\cdots+s_p}F)(zf(0))\\ &\quad\times\prod_{m=1}^p\left(\frac{1}{m!}UD_u^m(zf(u))\right)^{s_m}\\ &= z^{2p+1}\sum_{\sum_{i=1}^p is_i=p}\frac{1}{s_1!\cdots s_p!}(D^{s_1+\cdots+s_p}F)(zf(0))\prod_{m=1}^p\left(z\binom{2m}{m}\right)^{s_m}\\ &= z^{2p+1}\sum_{l=1}^p\frac{1}{l!}\binom{l}{s_1,\ldots,s_p}\prod_{m=1}^p\binom{2m}{m}^{s_m}z^lD^lF(z)\\ &= \sum_{l=1}^p\frac{\alpha_{l,p}}{l!}z^{l+2p+1}D^lF(z) = \Delta_pF(z)\end{aligned}$$

where we substituted $s_1+\cdots+s_p=l$ in the second line and split the sum according to the value of l and used $f(0)=1$ in the third line. □

Remark 6. More heuristically, we could argue in the following way: Regard $F(z)$ as a generating function of a tree-like structure where z marks the number of nodes. Then $F(z/\sqrt{1-4u})$ is the generating function where the nodes are substituted by a node and a sequence of "left-or-right" binary trees where the number of leaves is marked by u. Thus $[u^p]F(z/\sqrt{1-4u})$ is the generating function of those objects where the binary trees introduced by the substitution altogether contain exactly p leaves. The term z^{2p+1} accounts for introducing the $2p+1$ new nodes. This comes from counting the nodes of the binary trees coming from the substitution, adding an extra root for each of these trees and adding a new root to the total structure. This is precisely what Δ_p does.

[2] Faà di Bruno's formula is also stated in [9, p. 188, (III.24)], but unfortunately in the wrong form

$$\frac{h_n}{n!}=\sum_{k=1}^n\frac{f_k}{k!}\sum_{\sum_{j=1}^k j\ell_j=n,\ \sum_{j=1}^k\ell_j=k}\binom{k}{\ell_1,\ldots,\ell_k}\left(\frac{g_1}{1!}\right)^{\ell_1}\cdots\left(\frac{g_k}{k!}\right)^{\ell_k}$$

where $h_i=\frac{\mathrm{d}^i}{\mathrm{d}x^i}f(g(x))$, $f_i=\left(\frac{\mathrm{d}^i}{\mathrm{d}x^i}f\right)(g(x))$ and $g_i=\frac{\mathrm{d}^i}{\mathrm{d}x^i}g(x)$. The correct form is

$$\frac{h_n}{n!}=\sum_{k=1}^n\frac{f_k}{k!}\sum_{\sum_{j=1}^n j\ell_j=n,\ \sum_{j=1}^n\ell_j=k}\binom{k}{\ell_1,\ldots,\ell_n}\left(\frac{g_1}{1!}\right)^{\ell_1}\cdots\left(\frac{g_n}{n!}\right)^{\ell_n}.$$

or (in "non-exponential" form)

$$h_n=\sum_{k=1}^n f_k\sum_{\sum_{j=1}^n j\ell_j=n,\ \sum_{j=1}^n\ell_j=k}\frac{n!}{\ell_1!\ldots\ell_n!}\left(\frac{g_1}{1!}\right)^{\ell_1}\cdots\left(\frac{g_n}{n!}\right)^{\ell_n}.$$

The derivation of the differential equation of the generating function for $BCK(p)$-terms is a little more involved. Note that the differential operator Δ_p corresponds to p pointers from the root to some leaves. One is tempted to replace Δ_p in (3) by a sum of Δ_l's to take into account less than p pointers. But this is not entirely correct.

Proposition 7. *Let $F_p(z)$ be the generating function associated to $BCK(p)$-terms. Then $F_p(z) = Y(z/(1-z))$ where $Y(z)$ is the unique power series $Y(z) = \sum_{n\geqslant 0} Y_n z^n$ with nonnegative coefficients which satisfies*

$$Y(z) = \sum_{l=1}^{p} C_{l-1} z^{2l} + zY(z)^2 + \left(\sum_{l=1}^{p} \Delta_l\right) Y(z). \tag{7}$$

Proof. The (in some sense) minimal $BCK(p)$-terms are binary trees with at most p leaves and a unary root node pointing at all the leaves. This gives the first term on the right-hand side of (7).

Note that a unary node may also have zero pointers. A unary node with zero pointers which is not on top of the tree cannot be generated directly by a specification similar to (6). Therefore we first construct terms where each unary node has at least one pointer. Similar arguments as in the BCI case then lead directly to (7). Finally, we replace the edges by paths which exactly corresponds to the substitution $z \to z/(1-z)$. □

An alternative approach is to start with Motzkin trees with an additional root having pointers to all leaves as minimal structures. The terms with a unary root node can then be generated in the following way: Fix the number l of pointers we want to have at the root and then do an edge hitting process as in the BCI case. But instead of substituting the hit edges by sequences of left-or-right binary trees, use sequences of left-or-right Motzkin trees with an additional unary root node (corresponding to the nodes in the paths which substitute the hit edges) such that these trees have altogether l leaves. Recalling that on the level of generating functions edge hitting corresponds to applying a differential operator, we get in that way a differential equation for $F_p(z)$.

Proposition 8. *Let $M(z,u)$ denote the generating function of Motzkin trees where z marks the size (i.e. the total number of nodes) and u marks the number of leaves. This function is given by the unique power series solution of $M(z,u) = uz + zM(z,u) + zM(z,u)^2$, that is*

$$M(z,u) = \frac{1 - z - \sqrt{(1-z)^2 - 4uz^2}}{2z}. \tag{8}$$

Then $F_p(z)$ is given as the solution of

$$F_p(z) = z[u^p]\frac{M(z,u)}{1-u} + zF_p(z)^2 + z[u^p]\frac{1}{1-u} F_p\left(\frac{z}{1-2zM(z,u)}\right). \tag{9}$$

Proof. This is a direct consequence of the remarks above and Proposition 5. □

Let λ_n denote the number of closed lambda-terms and $\Lambda(z) = \sum_{n\geqslant 1} \lambda_n z^n$. Then we can use the two approaches presented above to find functional equations for $\Lambda(z)$.

Proposition 9. *Let $C(z) = (1-\sqrt{1-4z^2}\,)/2$ be the generating function associated to binary trees with an extra unary root node and counted by the number of nodes. Furthermore, let $\tilde{\Lambda}(z)$ be the power series solution of*

$$\tilde{\Lambda}(z) = C(z) + z\tilde{\Lambda}(z)^2 + z\tilde{\Lambda}\left(\frac{z}{1-2C(z)}\right) - z\tilde{\Lambda}(z). \tag{10}$$

Then $\Lambda(z) = \tilde{\Lambda}(z/(1-z))$. Moreover, we have

$$\Lambda(z) = zM(z,1) + z\Lambda(z)^2 + z\Lambda\left(\frac{z}{1-2zM(z,1)}\right). \tag{11}$$

Proof. To prove (10) we can proceed as in the proofs of Propositions 4 and 7 but allowing an unbounded number of edge hits instead. Thus, if $\tilde{\Lambda}(z)$ is the generating function associated to closed lambda-terms where each unary node carries at least one pointer, then

$$\tilde{\Lambda}(z) = \sum_{p\geqslant 1} C_{p-1}z^{2p} + z\tilde{\Lambda}(z)^2 + \mathcal{D}\tilde{\Lambda}(z)$$

where $\mathcal{D} = \sum_{p\geqslant 1}\Delta_p$. Now applying Proposition 5 yields (10). As in the BCK case, in order to create unary nodes carrying no pointers we replace the edges by paths which yields $\Lambda(z) = \tilde{\Lambda}(z/(1-z))$ and completes the proof of (10).

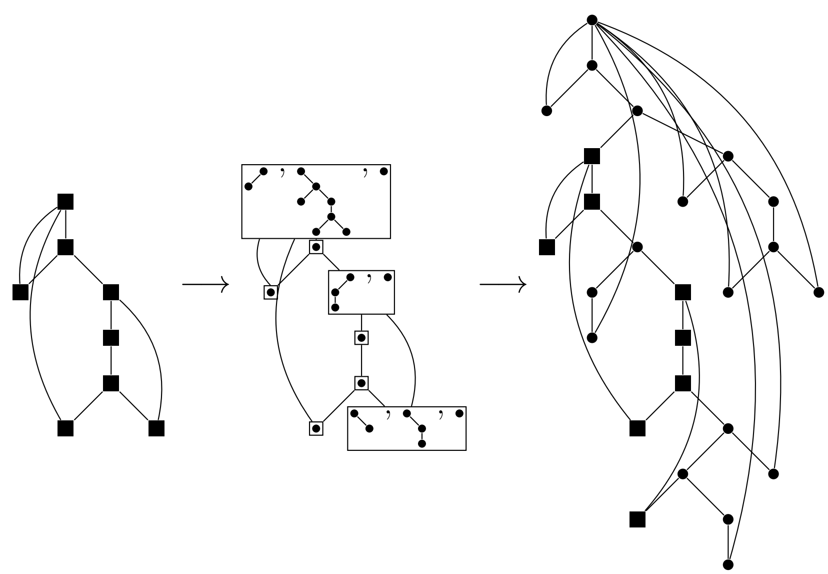

Figure 4: A step of the grafting expansion of a lambda-term

Alternatively, the lambda-terms with a unary root node can be created by starting with Motzkin trees with a unary node on top pointing to all leaves. These initial configurations are then expanded iteratively by substituting the edges by paths and attaching nodes, either left or right, which are (unary) roots of Motzkin trees, each binding all the leaves of its subtree. For an illustration of the expansion process, Figure 4 shows one step in this expansion process (not the initial one). Figure 5 presents one step of the reverse process. □

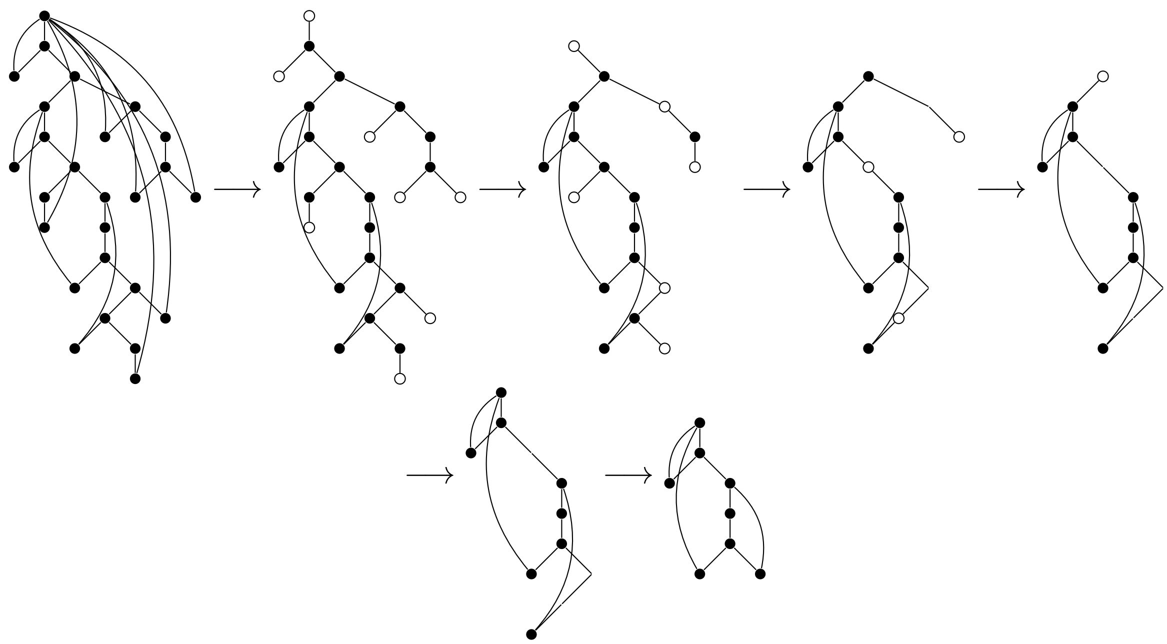

Figure 5: Finding the original closed lambda-term: first, the unary root node and all the leaves bound by the root are coloured white. Then delete the white nodes and colour all their neighbours white. Now continue recursively, where deletion of white unary nodes is done by removal and gluing the incident edges together.

4 The asymptotic number of $BCI(p)$-terms

Recall that $G_p(z) = \sum_{n\geqslant 1} g_{n(2p+1)-1} z^{n(2p+1)-1}$ is the generating function of the counting sequence of $BCI(p)$ terms. The function $G_p(z)$ satisfies the functional equation (3) which involves the differential operator Δ_p given by (4). Our goal is now to get a recurrence relation for the coefficients of $G_p(z)$.

Proposition 10. *The coefficients $g_{n(2p+1)-1}$ satisfy the recurrence relation*

$$g_{n(2p+1)-1} = \sum_{l=1}^{n-1} g_{l(2p+1)-1} g_{(n-1-l)(2p+1)-1} + Q_p(n-1) g_{(n-1)(2p+1)-1}, \text{ for } n \geqslant 2, \tag{12}$$

with initial condition $g_{(2p+1)-1} = C_{p-1}$ and where

$$Q_p(n) = \sum_{m=1}^{p} \alpha_{m,p} \binom{n(2p+1)-1}{m} \tag{13}$$

with $\alpha_{m,p}$ defined in (5).

Proof. Obvious, since the first term on the right-hand side of (3) only affects the case $n = 1$, the quadratic term is a Cauchy product and Δ_p is a linear combination of powers of the ordinary differential operator which acts on the coefficients of the power series exactly as shifting and multiplication by $Q_p(n-1)$ do. □

Lemma 11. *The polynomials $Q_p(n)$ can be represented more explicitly as*

$$Q_p(n) = 4^p \binom{\left(p+\frac{1}{2}\right)n + p - \frac{3}{2}}{p}.$$

Proof. Set $f(u) = 1/\sqrt{1-4u}$. It is easy to see that $\alpha_{m,p} = [u^p](f(u)-1)^m$ and that the coefficient on the right-hand side is zero if $m > p$. Thus we obtain

$$\begin{aligned} Q_p(n) &= \sum_{m=1}^{p} \binom{(2p+1)n-1}{m} \alpha_{m,p} \\ &= [u^p] \sum_{m \geqslant 1} \binom{(2p+1)n-1}{m} (f(u)-1)^m = [u^p] f(u)^{(2p+1)n-1} \\ &= 4^p \binom{\left(p+\frac{1}{2}\right)n + p - \frac{3}{2}}{p} \end{aligned}$$

and we are done. □

The key to the asymptotic analysis is a linearization of the differential equation which is possible due to the fast growth of the coefficients of $G_p(z)$. We start with an auxiliary result for fast growing sequences saying that in the Cauchy product only the extremal terms are asymptotically relevant:

Lemma 12. *Let $n_0 \in \mathbb{N}$ and $A(z) = \sum_{n \geqslant n_0} a_n z^n$ be a power series with positive coefficients (from index n_0 on). Assume that there exists $\sigma \geqslant 1$ with $a_{n+1}/a_n = \Omega(n^\sigma)$ as $n \to \infty$. Then $[z^n]A(z)^2 = 2a_{n_0} a_{n-n_0}(1 + O(n^{-\sigma}))$ as $n \to \infty$. If we want the second order term, we take the next two terms, and so on.*

Proof. Define $q_n = a_{n+1}/a_n$; then $1/q_n = O(n^{-\sigma})$. W.l.o.g. assume that n is odd. Then the coefficient of z^n in $A(z)^2$ is

$$\begin{aligned} \sum_{l=n_0}^{n-n_0} a_l a_{n-l} &= 2a_{n_0} a_{n-n_0} + 2 \sum_{l=1}^{\lfloor n/2 \rfloor - n_0} a_{n_0+l} a_{n-n_0-l} \\ &= 2a_{n_0} a_{n-n_0} \left(1 + \sum_{l=1}^{\lfloor n/2 \rfloor - n_0} \frac{q_{n_0} q_{n_0+1} \cdots q_{n_0+l-1}}{q_{n-n_0-1} q_{n-n_0-2} \cdots q_{n-n_0-l}} \right). \end{aligned}$$

In the case where n is even we have to subtract $\mathbf{1}_{\{n/2 \in \mathbb{N}\}} a_{n/2}^2$ on the r.-h. side.

The first term of the sum in the last line is $q_{n_0}/q_{n-n_0-1} = O((n-n_0-1)^{-\sigma}) = O(n^{-\sigma})$ (recall that n_0 is a constant). The further terms are of order $O(n^{-2\sigma})$ and there are not more than $\lfloor n/2 \rfloor$ of them. Thus the sum is of order $O(n^{1-2\sigma}) = O(n^{-\sigma})$. Hence

$$[z^n]A^2(z) = 2a_{n_0} a_{n-n_0}(1 + O(n^{-\sigma})) \sim [z^n] 2a_{n_0} z^{n_0} A(z).$$ □

We are now ready to derive bounds for the coefficients of $G_p(z)$.

Lemma 13. *Define $\phi_n = g_{n(2p+1)-1}$, $n \geqslant 1$. Then we have $\phi_{n+1}/\phi_n = \Omega(n^p)$ as $n \to \infty$.*

Proof. By (12) we have $\phi_0 = 0$, $\phi_1 = C_{p-1}$ and, for $n \geqslant 2$,

$$\phi_n = \sum_{l=1}^{n-2} \phi_l \phi_{n-1-l} + Q_p(n-1)\phi_{n-1}. \tag{14}$$

Thus $\phi_n \geqslant Q_p(n-1)\,\phi_{n-1}$. By Lemma 11 it is obvious that $Q_p(n)$ is a polynomial in n with leading term $\frac{2^p(2p+1)^p}{p!}n^p$ which implies the result. □

Corollary 14. *For fixed $p \geqslant 1$ and $n \to \infty$, the sum $\sum_{l=1}^{n-2} \phi_l \phi_{n-l-1}$ is asymptotically equal to $2\phi_1\phi_{n-2}(1 + O(1/n^p))$.*

Remark 15. The intuition behind the considerations above is as follows. From our study of $BCI(1)$ and from bounds already obtained (although for a different model) [7], we already know that the asymptotic behaviour of the number of lambda-terms widely differs from that of the number of trees: the significant increase in the number of lambda-terms of a given size when compared to Motzkin trees, i.e. the trees forming the underlying structure of lambda-terms, comes from the large number of ways of binding a leaf to unary nodes; indeed we are dealing here with directed acyclic graphs. Hence the rôle of the term G_p^2, which corresponds to the "purely binary tree-like" structure, is asymptotically negligible when compared to that of the differential term which captures the binding of leaves.

Remark 16. The exact differential equation for $G_p(z)$ is (3) whereas the arguments in Remark 15 show that we may work with the linearized[3] equation

$$L_p(z) = C_{p-1}z^{2p} + \Delta_p L_p(z). \tag{15}$$

The linearized equation has a combinatorial interpretation as well; indeed, it counts the number of structures $\mathcal{S}$ defined as follows: The smallest possible structures of $\mathcal{S}$ are precisely the smallest $BCI(p)$-terms, i.e., a unary root followed by a binary tree with $2p-1$ nodes (and pointing to all leaves of this binary tree). All terms in $\mathcal{S}$ have a unary node as their root. To construct larger terms, we add a new root and expand the sub-term below using the same edge hitting and expansion process as for $BCI(p)$-terms. Thus these terms may have binary nodes, but never as root.

Lemma 17. *For $p \geqslant 1$, the sequence $(\phi_n)_{n\geqslant 1}$ satisfies*

$$2\phi_1\phi_{n-2} \leqslant \sum_{l=1}^{n-2} \phi_l \phi_{n-1-l} \leqslant 2\phi_1\phi_{n-2} + (n-3)\phi_2\phi_{n-3}.$$

[3] This is not a linearization in a strict sense; we did not replace the quadratic term by a linear one, but only omitted it.

Proof. The lower bound is obvious: we just keep the first and the last term. Set $q_n = \phi_{n+1}/\phi_n$. To prove the upper bound, note that $(\phi_n)_{n\geqslant 1}$ is monotonically increasing and that for any $1 \leqslant i \leqslant \lfloor (n-3)/2 \rfloor$ we have

$$\phi_{2+i}\phi_{n-3-i} = \phi_2\phi_{n-3}\frac{q_2q_3\cdots q_{1+i}}{q_{n-2}q_{n-3}\cdots q_{n-1-i}} \geqslant \phi_2\phi_{n-3}. \qquad \square$$

Next we turn to the linearized equation (15).

Theorem 18. *Set* $\ell_{p,n} = [z^n]L_p(z)$ *where* L_p *is given by* (15). *Then, for fixed* p *and* $n \to \infty$,

$$\ell_{p,n} \sim B_p\beta_p^{n-1}n^{\gamma_p}(n-1)!^p$$

where

$$B_p = C_{p-1}\prod_{k=1}^{p}\frac{1}{\Gamma\left(1+\frac{2(p-k)-1}{2p+1}\right)} \tag{16}$$

$$= C_{p-1}\exp\left(-\frac{2p+1}{2}\int_1^2 \log(\Gamma(x))\,dx\right)\left(1+O\left(\frac{1}{p}\right)\right), \qquad \text{as } p\to\infty, \tag{17}$$

$$\approx C_{p-1}(1.0844375142\ldots)^{(2p+1)/2}\left(1+O\left(\frac{1}{p}\right)\right)$$

and

$$\beta_p = \frac{(4p+2)^p}{p!}, \qquad \gamma_p = \frac{p(p-2)}{2p+1}. \tag{18}$$

Proof. Equation (15) implies $\ell_{p,2p} = C_{p-1}$ and $\ell_{p,n} = Q_p(n-1)\ell_{p,n-2p-1}$ for $n > 2p$. Thus

$$\ell_{p,(2p+1)n-1} = C_{p-1}\prod_{j=1}^{n-1}Q_p(j)$$

$$= C_{p-1}\left(\frac{(4p+2)^p}{p!}\right)^{n-1}\prod_{k=1}^{p}\frac{\Gamma\left(n+\frac{2(p-k)-1}{2p+1}\right)}{\Gamma\left(1+\frac{2(p-k)-1}{2p+1}\right)}$$

$$= C_{p-1}\beta_p^{n-1}(n-1)!^p\prod_{j=1}^{n-1}\prod_{k=1}^{p}\left(1+\frac{2(p-k)-1}{2p+1}\cdot\frac{1}{j}\right). \tag{19}$$

Finally, note that, as $n\to\infty$,

$$C_{p-1}\prod_{j=1}^{n-1}\prod_{k=1}^{p}\left(1+\frac{2(p-k)-1}{2p+1}\cdot\frac{1}{j}\right) \sim B_pn^{\gamma_p}$$

which completes the proof. The asymptotic form of the constant B_p, given in (17), can be derived from Euler-McLaurin's formula. $\square$

p	a_p	A_p
2	1.048668...	0.981017...
3	1.0046726194...	2.19232485...
4	1.0006911656...	6.17349476...
5	1.0001221936...	19.2515312...

Table 1: The first few values of a_p.

Theorem 19. *For $p \geqslant 2$, the number of $BCI(p)$-terms of size $(2p+1)n-1$ is asymptotically*

$$A_p\, \beta_p^{n-1} n^{\gamma_p} (n-1)!^p$$

where β_p and γ_p are as in (18) and $A_p = a_p B_p$ with B_p as in (16) and $a_p = 1+O(1/(pe^p))$, as $p \to \infty$.

Remark 20. The first few values of the constants a_p and A_p appear in Table 1.

Remark 21. Applying Stirling's formula we get the alternative form

$$\bar{A}_p \bar{\beta}_p^{n-1} n^{\bar{\gamma}_p} n^{np}$$

where

$$\bar{\beta}_p = \frac{\beta_p}{e^p}, \qquad \bar{\gamma}_p = \frac{-5p}{4p+2}$$

and $\bar{A}_p = (2\pi/e^2)^{p/2} A_p$.

Proof. From the recurrence relation for ϕ_n, Equation (14), we have

$$\begin{aligned} \phi_n &= \phi_{n-1} Q_p(n-1) + \sum_{l=1}^{n-2} \phi_l \phi_{n-1-l} \\ &= \phi_{n-1} \left(Q_p(n-1) + \Gamma_{n-1}\right), \end{aligned}$$

with $\Gamma_{n-1} = \sum_{l=1}^{n-2} \phi_l \phi_{n-1-l}/\phi_{n-1}$ and $Q_p(n)$ defined in (13). Thus

$$\phi_n = \phi_1 \prod_{j=1}^{n-1} (Q_p(j) + \Gamma_j) = K_p(n) \phi_1 \prod_{j=1}^{n-1} Q_p(j)$$

where $K_p(n) = \prod_{j=1}^{n-1} \left(1 + \frac{\Gamma_j}{Q_p(j)}\right)$. For $p \geqslant 2$ we have $Q_p(n) = \Omega(n^p)$ and furthermore Corollary 14 gives $\Gamma_{n-1} = 2\phi_1 + O(1/n^p) = 2C_{p-1} + O(1/n^p)$. Hence the sequence $(K_p(n))_{n \geqslant 1}$ is convergent and we get

$$\phi_n = a_p C_{p-1} \left(\prod_{j=1}^{n-1} Q_p(j) \right) \left(1 + O\left(\frac{1}{n}\right) \right)$$

where $a_p = \lim_{n\to\infty} K_p(n)$. The product $C_{p-1}\prod_{j=1}^{n-1} Q_p(j)$ is already evaluated in (19), yielding the asymptotic behaviour of the solution of the linearized equation given in Theorem 18.

The difference between the linearization and the ϕ_n is hidden in the constant a_p. Thus we are left with the determination of a_p. We will confine ourselves with an asymptotic evaluation for $p \to \infty$.

First note that Lemma 11 immediately implies the inequality

$$Q_p(n) \geqslant \frac{2^p(2p+1)^p}{p!} n^p. \tag{20}$$

Now observe that $\Gamma_1 = 0$ and that by Lemma 17 we have $\Gamma_j \leqslant 2\phi_1 + (j-2)\phi_2\phi_{j-1}/\phi_j$. The quotient in the last term was already estimated in the proof of Lemma 13 by $\phi_{j-1}/\phi_j \leqslant 1/Q_p(j-1)$. Using this estimate as well as the inequality (20) we obtain (for $j > 1$)

$$\Gamma_j \leqslant 2\phi_1 + j\frac{\phi_2 p!}{2^p(2p+1)^p(j-1)^p} = 2C_{p-1} + j\frac{\phi_2 p!}{2^p(2p+1)^p(j-1)^p}.$$

Hence we get

$$\begin{aligned} a_p &= \prod_{j\geqslant 2}\left(1 + \frac{\Gamma_j}{Q_p(j)}\right) \\ &\leqslant \prod_{j\geqslant 2}\left(1 + \frac{2C_{p-1}p!}{2^p(2p+1)^p j^p} + \frac{\phi_2(p!)^2}{2^{2p}(2p+1)^{2p} j^{2p-1}}\right) \\ &\leqslant \prod_{j\geqslant 2}\left(1 + \frac{2C_{p-1}p!}{2^p(2p+1)^p j^p}\right)\prod_{j\geqslant 2}\left(1 + \frac{\phi_2(p!)^2}{2^{2p}(2p+1)^{2p}(j-1)^p j^{p-1}}\right). \end{aligned} \tag{21}$$

The two products above are of the form $\prod_{j\geqslant 2}\left(1+\frac{\varepsilon_p}{j^p}\right)$ and $\prod_{j\geqslant 2}\left(1+\frac{\varepsilon'_p}{(j-1)^p j^{p-1}}\right)$, resp., with $\varepsilon_p, \varepsilon'_p \to 0$ as $p \to \infty$. Thus we can easily estimate the first one by

$$\log\prod_{j\geqslant 1}\left(1+\frac{\varepsilon_p}{j^p}\right) = \sum_{j\geqslant 1}\sum_{k\geqslant 1}\frac{(-1)^{k-1}}{k}\frac{\varepsilon_p^k}{j^{pk}} = \sum_{k\geqslant 1}\frac{(-1)^{k-1}}{k}\varepsilon_p^k\zeta(pk). \tag{22}$$

Since $\zeta(x) = 1 + O(2^{-x})$ as $x \to \infty$, we obtain $\prod_{j\geqslant 1}\left(1+\frac{\varepsilon_p}{j^p}\right) = 1 + O(\varepsilon_p)$. Moreover, observe that $\prod_{j\geqslant 2}\left(1+\frac{\varepsilon'_p}{(j-1)^p j^{p-1}}\right) \leqslant \prod_{j\geqslant 1}\left(1+\frac{\varepsilon''_p}{j^p}\right)$ with $\varepsilon''_p = \varepsilon'_p/2^{p-1}$ which allows us to use (22) again.

Now turning to (21) we have, using $C_{p-1} \sim 4^{p-1}/\sqrt{\pi p^3}$,

$$\varepsilon_p = \frac{2C_{p-1}p!}{2^p(2p+1)^p} \sim \frac{1}{pe^p\sqrt{2e}}.$$

To estimate the second product in (21), observe that

$$\phi_2 = \phi_1 Q_p(1) = C_{p-1}4^p\binom{2p-1}{p} = 4^p(2p-1)C_{p-1}^2.$$

Thus we obtain

$$\begin{aligned}\varepsilon_p'' &= \frac{\phi_2(p!)^2}{2^{2p}(2p+1)^{2p}2^{p-1}} = \frac{(2p-1)2^{p-1}}{p}\left(\frac{2C_{p-1}p!}{2^p(2p+1)^p}\right)^2 \\ &\sim \frac{2^p}{pe^{2p}} = o\left(\frac{1}{pe^p}\right).\end{aligned}$$

This implies $a_p = 1 + O(1/(pe^p))$ which completes the proof. □

5 Closed lambda-terms

So far, we are unable to determine the asymptotic behaviour of λ_n. We will derive upper and lower estimates and a recurrence relation which allows an efficient computation of λ_n.

5.1 Estimates for λ_n

The number of $BCI(p)$-terms is certainly a lower bound, but using rather crude and elementary estimates a better bound can be obtained.

Theorem 22. *The number λ_n of closed lambda-terms of size n satisfies for every $\varepsilon > 0$ and for sufficiently large n the inequalities*

$$c_1\left(\frac{4n}{e\log n}\right)^{n/2}\frac{\sqrt{\log n}}{n} \leqslant \lambda_n \leqslant c_2\left(\frac{9(1+\varepsilon)n}{e\log n}\right)^{n/2}\frac{(\log n)^{n/(2\log n)}}{n^{3/2}}$$

where c_1, c_2 are some positive constants.

Proof. We determine the lower bound by counting particular lambda-terms of size n. Take a binary tree with n_f leaves and attach to its root a string of n_u unary nodes. Then connect the leaves to the unary nodes by pointers. Each such object is a closed lambda-term and there are $C_{n_f}n_u^{n_f}$ such terms. Note that $n_u = n + 1 - 2n_f$. Hence we obtain

$$\lambda_n \geqslant \sum_{n_u=1}^{n-1} C_{n_f} n_u^{(n+1-n_u)/2} \geqslant C_{\tilde{n}_f}\tilde{n}_u^{(n+1-\tilde{n}_u)/2}$$

where $\tilde{n}_u$ and $\tilde{n}_f$ are those values of n_u and n_f, respectively, where $n_u^{n_f}$ attains its maximum. The maximum is attained at $\tilde{n}_u = n/W(en)$ where $W(n)$ is Lambert's W-function defined implicitly by $W(n)e^{W(n)} = n$. It is easy to show that

$$W(en) = \log n - \log\log n + 1 + O\left(\frac{\log\log n}{\log n}\right).$$

This implies

$$\frac{n}{\log n} \leqslant \tilde{n}_u \leqslant \frac{n}{\log n - \log\log n}. \tag{23}$$

Hence we obtain

$$\begin{aligned}\tilde{n}_u^{\tilde{n}_f} &\geqslant \left(\frac{n}{\log n}\right)^{\tilde{n}_f} = \left(\frac{n}{\log n}\right)^{(n+1-\tilde{n}_u)/2} \\ &\geqslant \left(\frac{n}{\log n}\right)^{(n/2)\cdot(1-1/(\log n-\log\log n))+1/2} \\ &= \left(\frac{n}{\log n}\right)^{n/2}\sqrt{\frac{n}{\log n}}\exp\left(-\frac{n}{2(\log n-\log\log n)}(\log n-\log\log n)\right) \\ &= \left(\frac{n}{e\log n}\right)^{n/2}\sqrt{\frac{n}{\log n}}.\end{aligned}$$

The lower estimate now follows from $C_r \sim k_1 4^r/r^{3/2}$ $(r\to\infty)$ where k_1 is some positive constant.

For the upper estimate we construct a set of objects such that a proper subset corresponds to the set of all lambda-terms of size n. Take a Motzkin tree and add pointers such that each leaf is connected to an arbitrary unary node. Clearly, each lambda-term is generated in that way. But since leaf x might be bound to a unary node which is not on the path from x to the root, we generate also enriched trees which do not represent a lambda-term. Therefore we get the upper bound $\lambda_n \leqslant M_n \max n_u^{n_f}$ where M_n is the number of Motzkin trees with n vertices. As above we have $n_u = n/W(en)$. Now (23) implies that for sufficiently large n we have

$$\begin{aligned}n_u^{n_f} &\leqslant \left(\frac{n}{\log n-\log\log n}\right)^{\frac{n}{2}\left(1-\frac{1}{\log n}\right)} \\ &\leqslant \left(\frac{(1+\varepsilon)n}{\log n}\right)^{\frac{n}{2}}\left(\frac{n}{\log n}\right)^{-\frac{n}{2\log n}} \\ &= \left(\frac{(1+\varepsilon)n}{e\log n}\right)^{\frac{n}{2}}\exp\left(\frac{n\log\log n}{2\log n}\right)\end{aligned}$$

where we used $\log n/(1+\varepsilon) \leqslant \log n - \log\log n$ for sufficiently large n. Finally, the well known fact $M_r \sim k_2 3^r/r^{3/2}$ (as $r\to\infty$ and with some constant $k_2>0$) completes the proof. □

Remark 23. If $\bar{\lambda}_n$ is the number of closed lambda-terms where the sum of the number of unary nodes and the number of binary nodes equals n (so leaves do not contribute to the size), then David et al. [7] showed the following result for the growth rate of the counting sequence:

$$\left(\frac{(4-\epsilon)n}{\log n}\right)^{n-n/\log n} \leqslant \bar{\lambda}_n \leqslant \left(\frac{(12+\epsilon)n}{\log n}\right)^{n-n/3\log n}.$$

The underlying model is rather different from ours and so is the growth of the sequences. However, there is a relation: the exponential growth rates of of λ_n and $\tilde{\lambda}_n^2$ appear to be similar.

5.2 A recurrence relation

Equation (11) immediately implies that λ_n satisfies the recurrence relation

$$\lambda_n = M_{n-1} + \sum_{\ell+q=n-1} \lambda_\ell \lambda_q + \sum_{1\leqslant \ell \leqslant n-1} \delta_{n,\ell}\lambda_\ell \tag{24}$$

where $M_n = [z^n]M(z,1)$ is the number of Motzkin trees of size n and

$$\delta_{n,\ell} = [z^{n-1-\ell}]\frac{1}{(1-2zM(z))^\ell} = \sum_{r\geqslant 0}\binom{\ell-1+r}{\ell-1}\zeta_{n-\ell-1,r}$$

with $\zeta_{s,r} := [z^s](2zM(z))^r$. Note that $\zeta_{s,r}=0$ unless $s \geqslant 2r$ and thus

$$\delta_{n,\ell} = \sum_{r=0}^{\lfloor (n-\ell-1)/2\rfloor}\binom{\ell-1+r}{\ell-1}\zeta_{n-\ell-1,r}.$$

By Lagrange inversion we obtain

$$\zeta_{s,r} = 2^r[z^{s-r}]M(z)^r = 2^r\frac{r}{s-r}\sum_{a,b,c\colon\, b+2c=s-2r}\binom{s-r}{a,b,c}$$

which gives after a few computations

$$\delta_{n,\ell} = \begin{cases} \displaystyle\sum_{t=0}^{\left\lfloor \frac{n-\ell-1}{2}\right\rfloor}\sum_{r=0}^{t}\frac{r2^r\binom{\ell-1+r}{r}(n-\ell-2-r)!}{t!\,(t-r)!\,(n-\ell-1-2\,t)!} & \text{if } 1\leqslant \ell < n-1,\\ 1 & \text{if } \ell = n-1. \end{cases} \tag{25}$$

Now, consider the inner sum and set $b_{n,\ell,t} := \sum_{r=0}^{t}\frac{r2^r\binom{\ell-1+r}{r}(n-\ell-2-r)!}{t!\,(t-r)!\,(n-\ell-1-2\,t)!}$. This sum is amenable to creative telescoping (see [21]) which yields a system of two recurrences of order one for the multi-index sequence $(b_{n,\ell,t})_{n,\ell,t\geqslant 0}$:

$$\begin{aligned}&\left(-\ell^2-2nt-2\ell t-\ell-n+n^2\right)b_{n,\ell,t}+\left(2\ell t-2\,n\ell+2\ell^2+4\ell\right)b_{n,\ell+1,t}\\&\quad+\left(-4t^2-2t+4nt-4\ell t-n^2+2n\ell-\ell+n-\ell^2\right)b_{n+1,\ell,t}=0\end{aligned}$$

and

$$\begin{aligned}&(2n-t-2)\,(n-\ell-2t-2)\,(n-\ell-2t-1)\,b_{\ell,n,t}-t\,(t+1)\,(n-\ell-t-2)\,b_{\ell,n,t+1}\\&\quad-(n-\ell-2t-2)(n-\ell-2t-1)\,(n-\ell-2t)\,b_{\ell,n+1,t}=0.\end{aligned}$$

with the initial conditions given by the sum representation of $b_{n,\ell,t}$. This system can be solved explicitly and we get

$$b_{n,\ell,t} = \frac{2\ell}{t}\cdot\frac{\Gamma\,(n-\ell-2)\ {}_2F_1(-t+1,\ell+1;-n+\ell+3;2)}{\Gamma\,(t)^2\,\Gamma\,(n-\ell-2t)}$$

where ${}_2F_1$ denotes the Gauss hypergeometric function defined by

$$ {}_2F_1(a, b; c; z) = \sum_{k\geqslant 0} \frac{(a)_k (b)_k z^k}{(c)_k k!} \text{ if } |z| < 1 \text{ or } |z| = 1 \text{ and } \Re(c-a-b) > 0, $$

where $(a)_k$ denotes the falling factorial $(a)_k = a(a-1)\cdots(a-k+1)$. There are several continuation formulas to other domains of the complex plane. In our case one could for instance use

$$ \begin{aligned} {}_2F_1(a, b; c; z) = &\frac{\Gamma(b-a)\Gamma(c)(-z)^{-a}}{\Gamma(b)\Gamma(c-a)} \sum_{k\geqslant 0} \frac{(a)_k (a-c+1)_k z^{-k}}{(a-b+1)_k k!} \\ &+ \frac{\Gamma(a-b)\Gamma(c)(-z)^{-b}}{\Gamma(a)\Gamma(c-b)} \sum_{k\geqslant 0} \frac{(b)_k (b-c+1)_k z^{-k}}{(b-a+1)_k k!} \text{ if } |z| > 1 \text{ and } a-b \notin \mathbb{Z}, \end{aligned} $$

but the important issue here is not the particular representation but rather the recurrence relations satisfied by ${}_2F_1(a, b; c; z)$ (see e.g. [10] for further reading). Indeed these properties make it useful in computer algebra systems. Now, using creative telescoping again, this time for $\delta_{n,\ell} = \sum_{t=0}^{\lfloor \frac{n-\ell-1}{2} \rfloor} b_{n,\ell,t}$, we can also obtain a system of two D-finite recurrences for $\delta_{n,\ell}$ (fully automatically with computer algebra packages):

$$ \begin{aligned} &(n-\ell)(n+1-\ell)(n-2\ell-2)\,\delta_{n+2,\ell} - (n-\ell)\left(2n^2 - 6n\ell - 5n + 2\ell^2 + 3\ell + 1\right)\delta_{n+1,\ell} \\ &\quad - (n-1)\left(3n^2 - 2n\ell + n - \ell^2 - 9\ell - 8\right)\delta_{n,\ell} + 20\,(n-1)\,\ell\,(\ell+1)\,\delta_{n,\ell+2} \\ &\quad + 2\,(n-1)(5n - 9\ell - 12)\,\ell\delta_{n,\ell+1} = 0 \end{aligned} $$

and

$$ \begin{aligned} (n-\ell)(\ell-n-1)\,\delta_{n+2,\ell} + (n-\ell)(2n-\ell)\,\delta_{n+1,\ell} - \ell\,(n-1)\,\delta_{n+1,\ell+1} \\ - 4\ell\,(n-1)\,\delta_{n,\ell+1}\,(n-1) + (3n - 2\ell + 1)\,\delta_{n,\ell} = 0 \end{aligned} \tag{26} $$

with initial conditions $\delta_{n,n} = 0$, $\delta_{n,n-1} = 1$, $\delta_{n,n-2} = 0$ for $n \geqslant 2$. Unfortunately, this equation seems not to admit an explicit solution in terms of classical special functions. Nevertheless, there exist powerful computer algebra methods for D-finite recurrences which are implemented in standard Maple packages, for instance. By means of these methods it is possible to use such (at first sight complicated looking) expressions like (26) for a very efficient computation of the values $\delta_{n,\ell}$ (see e.g. [4]). Experiments on a 1.5 GHz notebook using Maple showed that the first 1000 terms of $(\lambda_n)_{n\geqslant 1}$ can be computed in a few seconds. This was not possible with other approaches like for instance using the functional equation of the generating function which was given in [3].

6 Conclusion and outlook

The motivation for our analysis was the enumeration of closed lambda-terms. Since the problem seems hard, we treated the subclass of $BCI(p)$-terms which imposes quite a restriction on the degrees of freedom in binding variables by quantifiers. Thus we expected the set of $BCI(p)$-terms to be small in comparison to the set of closed lambda-terms. Our results verify and quantify this. Moreover, they show that the restriction is weaker than bounding the unary height, i.e., the maximal number of unary nodes on a root-to-leaf path. Indeed, if the unary height is bounded by L, then the asymptotic number of terms of size n is in general $Cn^{-3/2}\rho^n$; i.e. the asymptotic behaviour is like that of the number of Motzkin trees. Only for particular values of L, the asymptotic behaviour becomes $Cn^{-5/4}\rho^n$ (see [3]). This behaviour changes if the condition on the unary height is replaced by a condition on the number of pointers per unary node (as in the $BCI(p)$ case) or dropped completely (closed lambda-terms). So these structures are indeed different from tree-like structures; their counting sequences grow much faster than that of Motzkin trees. So we can conclude that the enumeration of $BCI(p)$-terms is not only of interest in its own right, but also more closely related to the original counting problem than to tree enumeration.

Since the union of all the sets of $BCK(p)$-terms, $p = 1, 2, \ldots$, is precisely the set of closed lambda-terms, one might be tempted to approach the problem of determining the asymptotic behaviour of λ_n via the number of $BCK(p)$-terms and letting $p \to \infty$. For performing such a limit we needed precise and uniform asymptotics for the number of $BCK(p)$-terms. Unfortunately, the asymptotic computation of the number of $BCK(p)$-terms turns out to be much more involved than that of the number of $BCI(p)$-terms. A precise analysis of the BCK case is beyond the scope of this paper and will be the topic of a forthcoming paper. Here we discuss only briefly how to attack this problem.

The differential equation (9) implies a recurrence relation for the coefficients of $F_p(z)$. This can be linearized in a similar fashion as we did in the BCI case (essentially Lemmas 12-17). The next step will be showing upper and lower estimates for $f_n := [z^n]F_p(z)$. This enables us to identify the asymptotically dominant term in the recursion which yields a rough information on the growth of f_n.

The task is now to find the asymptotic behaviour of the correct solution. The growth rate of the coefficients tells us that the Borel transform $\hat{F}_p(z)$ of the generating function $F_p(z)$ must grow exponentially in z. This indicates that $\hat{F}_p(z)$ is Hayman-admissible (*cf.* [14]) and therefore a saddle point analysis applies and eventually yields the asymptotic number of $BCK(p)$-terms.

When studying not only the size but further properties of $BCK(p)$-terms by means of multivariate generating functions, the above remarks suggest that these functions will be (multivariate) Hayman-admissible such that a multivariate saddle point method applies (*cf.* [11]).

As in the case of closed lambda-terms, the functional equation (9) corresponds to a recurrence relation of the form (24). The only difference is that $\delta_{n,l}$ in (25) has to be replaced by

$$\delta_{n,l} = \begin{cases} \displaystyle\sum_{t=0}^{\min\left(p, \left\lfloor \frac{n-l-1}{2} \right\rfloor\right)} \sum_{r=0}^{t} \frac{r 2^r \binom{l-1+r}{r} (n-l-2-r)!}{t!\,(t-r)!\,(n-l-1-2t)!} & \text{if } 1 \leqslant l < n-1, \\ 1 & \text{if } l = n-1. \end{cases}$$

Similarly as before, this gives rise to a system of D-finite recursions.

Acknowledgement. We thank Marek Zaionc for triggering our interest in the subject and for numerous fruitful discussions about it. Furthermore we thank an anonymous referee for the careful reading of the manuscript and pointing out several imprecisions.

References

[1] Henk P. Barendregt. *The lambda calculus. Its syntax and semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, revised edition, 1984.

[2] O. Bodini, D. Gardy, and A. Jacquot. Asymptotics and random sampling for BCI and BCK lambda terms. *Theor. Comput. Sci.*, 2013. doi:10.1016/j.tcs.2013.01.008.

[3] Olivier Bodini, Danièle Gardy, and Bernhard Gittenberger. Lambda-terms of bounded unary height. In *ANALCO, workshop on ANALytic COmbinatorics*, San Francisco (USA), January 2011.

[4] Alin Bostan, Frédéric Chyzak, Bruno Salvy, Grégoire Lecerf, and Éric Schost. Differential equations for algebraic functions. In *ISSAC 2007*, pages 25–32. ACM, New York, 2007.

[5] Alonzo Church. An Unsolvable Problem of Elementary Number Theory. *Amer. J. Math.*, 58(2):345–363, 1936.

[6] Louis Comtet. *Advanced combinatorics*. D. Reidel Publishing Co., Dordrecht, enlarged edition, 1974.

[7] René David, Katarzyna Grygiel, Jakub Kozik, Christophe Raffalli, Guillaume Theyssier, and Marek Zaionc. Asymptotically almost all λ-terms are strongly normalizing. *Log. Methods Comput. Sci.*, 9(1):1:02, 30, 2013.

[8] Ph. de Groote, editor. *The Curry-Howard isomorphism*, volume 8 of *Cahiers du Centre de Logique [Reports of the Center of Logic]*. Academia-Erasme, Louvain, 1995.

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

[10] George Gasper and Mizan Rahman. *Basic hypergeometric series*, volume 96 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, second edition, 2004. With a foreword by Richard Askey.

[11] Bernhard Gittenberger and Johannes Mandlburger. Hayman admissible functions in several variables. *Electron. J. Combin.*, 13(1):Research Paper 106, 29 pp. (electronic), 2006.

[12] Katarzyna Grygiel, PawełIdziak, and Marek Zaionc. How big is BCI fragment of BCK logic. *J. Logic Comput.*, 23(3):673–691, 2013.

[13] Katarzyna Grygiel and Pierre Lescanne. Counting and generating lambda terms. *J. Functional Programming*, 23(5): 594–628, 2013.

[14] Walter K. Hayman. A generalisation of Stirling's formula. *J. Reine Angew. Math.*, 196:67–95, 1956.

[15] Yasuyuki Imai and Kiyoshi Iséki. Corrections to: "On axiom systems of propositional calculi. I". *Proc. Japan Acad.*, 41:669, 1965.

[16] Yasuyuki Imai and Kiyoshi Iséki. On axiom systems of propositional calculi I. *Proc. Japan Acad.*, 41:436–439, 1965.

[17] Kiyoshi Iséki and Shôtarô Tanaka. An introduction to the theory of BCK-algebras. *Math. Japon.*, 23(1):1–26, 1978/79.

[18] Stephen C. Kleene. A Theory of Positive Integers in Formal Logic. Part I. *Amer. J. Math.*, 57(1):153–173, 1935.

[19] Stephen C. Kleene. A Theory of Positive Integers in Formal Logic. Part II. *Amer. J. Math.*, 57(2):219–244, 1935.

[20] Morten Heine Sørensen and Paweł Urzyczyn. *Lectures on the Curry-Howard isomorphism.* Amsterdam: Elsevier, 2006.

[21] Doron Zeilberger. The method of creative telescoping. *J. Symbolic Comput.*, 11(3):195–204, 1991.