On Repetitive Right Application of B-Terms

Mirai Ikebuchi
Massachusetts Institute of Technology, Cambridge, MA, USA
https://mir-ikbch.github.io/
ikebuchi@mit.edu

Keisuke Nakano[1]
Tohoku University, Sendai, Miyagi, Japan
http://www.riec.tohoku.ac.jp/~ksk/
k.nakano@acm.org

Abstract

B-terms are built from the B combinator alone defined by $B \equiv \lambda f.\lambda g.\lambda x.f\ (g\ x)$, which is well-known as a function composition operator. This paper investigates an interesting property of B-terms, that is, whether repetitive right applications of a B-term cycles or not. We discuss conditions for B-terms to have and not to have the property through a sound and complete equational axiomatization. Specifically, we give examples of B-terms which have the property and show that there are infinitely many B-terms which do not have the property. Also, we introduce a canonical representation of B-terms that is useful to detect cycles, or equivalently, to prove the property, with an efficient algorithm.

2012 ACM Subject Classification Theory of computation → Rewrite systems

Keywords and phrases Combinatory logic, B combinator, Lambda calculus

Digital Object Identifier 10.4230/LIPIcs.FSCD.2018.18

Acknowledgements The authors would like to thank the anonymous reviewers for their valuable comments that improved the manuscript.

1 Introduction

The 'bluebird' combinator $B = \lambda f.\lambda g.\lambda x.f\ (g\ x)$ is well-known [10] as a bracketing combinator or composition operator, which has a principal type $(\alpha \to \beta) \to (\gamma \to \alpha) \to \gamma \to \beta$. A function $B\ f\ g$ (also written as $f \circ g$) takes a single argument x and returns the term $f\ (g\ x)$.

In the general case that g takes n arguments, the composition of f and g, defined by $\lambda x_1.\cdots\lambda x_n.f\ (g\ x_1\ \ldots\ x_n)$, can be expressed as $B^n\ f\ g$ where e^n is the n-fold composition $\underbrace{e \circ \cdots \circ e}_{n}$ of the function e, or equivalently given by $e^n\ x = \underbrace{e\ (\ldots (e}_{n}\ x))$ [1, Definition 2.1.9]. We call *n-argument composition* for the generalized composition represented by B^n.

Now we consider the 2-argument composition expressed as $B^2 = \lambda f.\lambda g.\lambda x.\lambda y.\ f\ (g\ x\ y)$. From the definition, we have $B^2 = B \circ B = B\ B\ B$. Note that function application is considered left-associative, that is, $f\ a\ b = (f\ a)\ b$. Thus B^2 is expressed as a term in which all applications nest to the left, never to the right. We call such terms *flat* [9]. We write $X_{(k)}$ for the flat term defined by $\underbrace{X\ X\ X\ \ldots\ X}_{k} = \underbrace{(\ldots((X\ X)\ X)\ldots)}_{k}\ X$. Using this notation, we can write $B^2 = B_{(3)}$.

[1] This work was partially supported by JSPS KAKENHI Grant Number JP25730002 and JP17K00007.

3rd International Conference on Formal Structures for Computation and Deduction (FSCD 2018).
Editor: Hélène Kirchner; Article No. 18; pp. 18:1–18:15
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$$\begin{tikzcd}[arrows=dash]
B_{(1)} \arrow[r] & B_{(2)} \arrow[r] & B_{(3)} \arrow[r] & B_{(4)} \arrow[r] & B_{(5)} \arrow[r] & B_{(6)}\,(= B_{(10)} = B_{(14)} = \dots) \arrow[r] \arrow[d] & B_{(7)}\,(= B_{(11)} = B_{(15)} = \dots) \arrow[d] \\
 & & & & & B_{(9)}\,(= B_{(13)} = B_{(17)} = \dots) \arrow[r] & B_{(8)}\,(= B_{(12)} = B_{(16)} = \dots)
\end{tikzcd}$$

Figure 1 ρ-property of the B combinator.

Okasaki [9] investigated facts about flatness. For example, he shows that there is no universal combinator X that can represent any combinator by $X_{(k)}$ with some k. We shall delve into the case of $X = B$. Consider the n-argument composition operator B^n. We have already seen that B^2 can be written by the flat term $B_{(3)}$. For $n = 3$, we can also check $B^3 = B\ B\ B\ B\ B\ B\ B\ B = B_{(8)}$ by repeating β-reduction for $B_{(8)}\ f\ g\ x\ y\ z = f\ (g\ x\ y\ z)$. How about the 4-argument composition B^4? In fact, there is no integer k such that $B^4 = B_{(k)}$ with respect to $\beta\eta$-equality. Moreover, for any $n > 3$, there does not exist k such that $B^n = B_{(k)}$. This surprising fact is proved by a quite simple method; listing all $B_{(k)}$s for $k = 1, 2, \dots$ and checking that none of them is equivalent to B^n. An easy computation gives $B_{(6)} = B_{(10)} = \lambda x.\lambda y.\lambda z.\lambda w.\lambda v.\ x\ (y\ z)\ (w\ v)$, and hence $B_{(i)} = B_{(i+4)}$ for every $i \geq 6$. Then, by computing $B_{(k)}$s only for $k \in \{1, 2, \dots, 6\}$, we can check that $B_{(k)}$ is not $\beta\eta$-equivalent to B^n with $n > 3$ for $k \in \{1, 2, \dots\}$. Thus we conclude that there is no integer k such that $B^n = B_{(k)}$.

This is the starting point of our research. We call *ρ-property* for this "periodicity" on combinatory terms. More precisely, we say that a combinator X has ρ-property if there exist two distinct integers i and j such that $X_{(i)} = X_{(j)}$. In this case, we have $X_{(i+k)} = X_{(j+k)}$ for any $k \geq 0$ (à la *finite monogenic semigroup* [7]). Fig. 1 shows a computation graph of $B_{(k)}$. The ρ-property is named after the shape of the graph.

This paper discusses the ρ-property of combinatory terms, particularly terms built from B alone. We call such terms *B-terms* and $\mathbf{CL}(B)$ denotes the set of all B-terms. For example, the B-term $B\ B$ enjoys the ρ-property with $(B\ B)_{(52)} = (B\ B)_{(32)}$ and so does $B\ (B\ B)$ with $(B\ (B\ B))_{(294)} = (B\ (B\ B))_{(258)}$ as reported in [8]. Several combinators other than B-terms can be found to enjoy the ρ-property, for example, $K = \lambda x.\lambda y.x$ and $C = \lambda x.\lambda y.\lambda z.\ x\ z\ y$ because of $K_{(3)} = K_{(1)}$ and $C_{(4)} = C_{(3)}$. They are less interesting in the sense that the cycle starts immediately and its size is very small, comparing with B-terms like $B\ B$ and $B\ (B\ B)$. As we will see later, $B\ (B\ (B\ (B\ (B\ (B\ B)))))(\equiv B^6\ B)$ has the ρ-property with the cycle of the size more than 3×10^{11} which starts after more than 2×10^{12} repetitive right applications. This is why the ρ-property of B-terms is intensively discussed in the present paper.

The contributions of the paper are two-fold. One is to give a characterization of $\mathbf{CL}(B)$ (Section 3) and another is to provide a sufficient condition for the ρ-property and anti-ρ-property of B-terms (Section 4). In the former, we introduce a canonical representation of B-terms and establish a sound and complete equational axiomatization for $\mathbf{CL}(B)$. In the latter, the ρ-property of $B^n B$ with $n \leq 6$ is shown with an efficient algorithm and the anti-ρ-property for B-terms of particular forms is proved.

2 ρ-property of terms

The ρ-property of combinator X is that $X_{(i)} = X_{(j)}$ holds for some $i > j \geq 1$. We adopt $\beta\eta$-equality of corresponding λ-terms for the equality of combinatory terms in this paper. We could use other equality, for example, induced by the axioms of combinatory logic. The choice of equality is not essential here, e.g., $B_{(9)}$ and $B_{(13)}$ are equal even up to the combinatory axiom of B, as well as $\beta\eta$-equality. Furthermore, for simplicity, we only deal with the case

$$\begin{aligned}
\rho(B^0 B) &= (6, 4) & \rho(B^4 B) &= (191206, 431453)\\
\rho(B^1 B) &= (32, 20) & \rho(B^5 B) &= (766241307, 234444571)\\
\rho(B^2 B) &= (258, 36) & \rho(B^6 B) &= (2641033883877, 339020201163)\\
\rho(B^3 B) &= (4240, 5796)
\end{aligned}$$

Figure 2 ρ-property of B-terms in a particular form.

where $X_{(n)}$ is normalizable for all n. If $X_{(n)}$ is not normalizable, it is much more difficult to check equivalence with the other terms. This restriction does not affect results of the paper because all B-terms are normalizing.

Let us write $\rho(X) = (i, j)$ if a combinator X has the ρ-property due to $X_{(i)} = X_{(i+j)}$ with minimum positive integers i and j. For example, we have $\rho(B) = (6, 4)$, $\rho(C) = (3, 1)$, $\rho(K) = (1, 2)$ and $\rho(I) = (1, 1)$. Besides them, several combinators introduced in Smullyan's book [10] have the ρ-property:

$$\begin{aligned}
\rho(D) &= (32, 20) && \text{where } D = \lambda x.\lambda y.\lambda z.\lambda w.x\ y\ (z\ w)\\
\rho(F) &= (3, 1) && \text{where } F = \lambda x.\lambda y.\lambda z.z\ y\ x\\
\rho(R) &= (3, 1) && \text{where } R = \lambda x.\lambda y.\lambda z.y\ z\ x\\
\rho(T) &= (2, 1) && \text{where } T = \lambda x.\lambda y.y\ x\\
\rho(V) &= (3, 1) && \text{where } V = \lambda x.\lambda y.\lambda z.z\ x\ y.
\end{aligned}$$

Except the B and D $(= B\ B)$ combinators, the property is 'trivial' in the sense that the loop starts early and the size of cycle is very small.

On the other hand, the combinators $S = \lambda x.\lambda y.\lambda z.x\ z\ (y\ z)$ and $O = \lambda x.\lambda y.y\ (x\ y)$ in the book do not have the ρ-property for reason (A), which is illustrated by

$$\begin{aligned}
S_{(2n+1)} &= \lambda x.\lambda y.\underbrace{x\ y\ (x\ y\ (\ldots(x\ y}_{n}\ (\lambda z.x\ z\ (y\ z)))\ldots)),\\
O_{(n+1)} &= \lambda x.\underbrace{x\ (x\ (\ldots(x}_{n}\ (\lambda y.y\ (x\ y)).
\end{aligned}$$

The definition of the ρ-property is naturally extended from single combinators to terms obtained by combining several combinators. We found by computation that several B-terms, built from the B combinator alone, have a nontrivial ρ-property as shown in Fig. 2. The detail will be shown in Section 4.

3 Checking equivalence of B-terms

The set of all B-terms, $\mathbf{CL}(B)$, is closed under application by definition, that is, the repetitive right application of a B-term always generates a sequence of B-terms. Hence, the ρ-property can be decided by checking 'equivalence' among generated B-terms, where the equivalence should be checked through $\beta\eta$-equivalence of their corresponding λ-terms in accordance with the definition of the ρ-property. It would be useful if we have a fast algorithm for deciding equivalence over B-terms.

In this section, we give a characterization of the B-terms to efficiently decide their equivalence. We introduce a method for deciding equivalence of B-terms without calculating the corresponding λ-terms. To this end, we first investigate equivalence over B-terms with

$$B\ x\ y\ z = x\ (y\ z) \tag{B1}$$
$$B\ (B\ x\ y) = B\ (B\ x)\ (B\ y) \tag{B2}$$
$$B\ B\ (B\ x) = B\ (B\ (B\ x))\ B \tag{B3}$$

Figure 3 Equational axiomatization for B-terms

examples and then present an equation system as a characterization of B-terms so as to decide equivalence between two B-terms. Based on the equation system, we introduce a canonical representation of B-terms. The representation makes it easy to observe the growth caused by repetitive right application of B-terms, which will be later used for proving the anti-ρ-property of B^2. We believe that this representation will be helpful to prove the ρ-property or the anti-ρ-property for the other B-terms.

3.1 Equivalence over B-terms

Two B-terms are said *equivalent* if their corresponding λ-terms are $\beta\eta$-equivalent. For instance, $B\ B\ (B\ B)$ and $B\ (B\ B)\ B\ B$ are equivalent. This can be easily shown by the definition $B\ x\ y\ z = x\ (y\ z)$. For another (non-trivial) instance, $B\ B\ (B\ B)$ and $B\ (B\ (B\ B))\ B$ are equivalent. This is illustrated by the fact that they are equivalent to $\lambda x.\lambda y.\lambda z.\lambda w.\lambda v.x\ (y\ z)\ (w\ v)$ where B is replaced with $\lambda x.\lambda y.\lambda z.\ x\ (y\ z)$ or the other way around at the $=_\beta$ equation. Similarly, we cannot show equivalence between two B-terms, $B\ (B\ B)\ (B\ B)$ and $B\ (B\ B\ B)$, without long calculation. This kind of equality makes it hard to investigate the ρ-property of B-terms. To solve this annoying issue, we will later introduce a canonical representation of B-terms.

3.2 Equational axiomatization for B-terms

Equality between two B-terms can be effectively decided by an equation system. Figure 3 shows a sound and complete equation system as described in the following theorem.

▶ **Theorem 1.** *Two B-terms are $\beta\eta$-equivalent if and only if their equality is derived by equations (B1), (B2), and (B3).*

The proof of the “if” part, which corresponds to the soundness of the equation system (B1), (B2), and (B3), is given here. We will later prove the “only if” part with the uniqueness of the canonical representation of B-terms.

Proof. Equation (B1) is immediate from the definition of B. Equations (B2) and (B3) are shown by

$$\begin{aligned}
B\ (B\ e_1\ e_2) &= \lambda x.\lambda y.\ B\ (B\ e_1\ e_2)\ x\ y \\
&= \lambda x.\lambda y.\ B\ e_1\ e_2\ (x\ y) \\
&= \lambda x.\lambda y.\ e_1\ (e_2\ (x\ y)) \\
&= \lambda x.\lambda y.\ e_1\ (B\ e_2\ x\ y) \\
&= \lambda x.\ B\ e_1\ (B\ e_2\ x) \\
&= B\ (B\ e_1)\ (B\ e_2)
\end{aligned}$$

$$\begin{aligned}
B\ B\ (B\ e_1) &= \lambda x.\ B\ B\ (B\ e_1)\ x \\
&= \lambda x.\ B\ (B\ e_1\ x) \\
&= \lambda x.\lambda y.\lambda z.\ B\ e_1\ x (y\ z) \\
&= \lambda x.\lambda y.\lambda z.\ e_1\ (x\ (y\ z)) \\
&= \lambda x.\lambda y.\lambda z.\ e_1\ (B\ x\ y\ z) \\
&= \lambda x.\lambda y.\ B\ e_1\ (B\ x\ y) \\
&= \lambda x.\ B\ (B\ e_1)\ (B\ x) \\
&= B\ (B\ (B\ e_1))\ B
\end{aligned}$$

where the α-renaming is implicitly used. ◀

Equation (B2) has been employed by Statman [12] to show that no $B\omega$-term can be a fixed-point combinator where $\omega = \lambda x.x\ x$. This equation exposes an interesting feature of the B combinator. Write equation (B2) as

$$B\ (e_1 \circ e_2) = (B\ e_1) \circ (B\ e_2) \tag{B2'}$$

by replacing every B combinator with $\circ$ infix operator if it has exactly two arguments. The equation is a distributive law of B over $\circ$, which will be used to obtain the canonical representation of B-terms. Equation (B3) is also used for the same purpose as the form of

$$B \circ (B\ e_1) = (B\ (B\ e_1)) \circ B. \tag{B3'}$$

We also have a natural equation $B\ e_1\ (B\ e_2\ e_3) = B\ (B\ e_1\ e_2)\ e_3$ which represents associativity of function composition, i.e., $e_1 \circ (e_2 \circ e_3) = (e_1 \circ e_2) \circ e_3$. This is shown with equations (B1) and (B2) by

$$B\ e_1\ (B\ e_2\ e_3) = B\ (B\ e_1)\ (B\ e_2)\ e_3 = B\ (B\ e_1\ e_2)\ e_3.$$

3.3 Canonical representation of B-terms

To decide equality between two B-terms, it does not suffice to compute their normal forms under the definition of B, $B\ x\ y\ z \to x\ (y\ z)$. This is because two distinct normal forms may be equal up to $\beta\eta$-equivalence, e.g., $B\ B\ (B\ B)$ and $B\ (B\ (B\ B))\ B$. We introduce a canonical representation of B-terms, which makes it easy to check equivalence of B-terms. We will eventually find that for any B-term e there exists a unique finite non-empty weakly-decreasing sequence of non-negative integers $n_1 \geq n_2 \geq \cdots \geq n_k$ such that e is equivalent to $(B^{n_1}B) \circ (B^{n_2}B) \circ \cdots \circ (B^{n_k}B)$. Ignoring the inequality condition gives *polynomials* introduced by Statman [12]. We will use these *decreasing polynomials* for our canonical representation as presented later. A similar result is found in [4].

First, we explain how this canonical form is obtained from a B-term. We only need to consider B-terms in which every B has at most two arguments. One can easily reduce the arguments of B to less than three by repeatedly rewriting occurrences of $B\ e_1\ e_2\ e_3\ e_4\ \ldots\ e_n$ into $e_1\ (e_2\ e_3)\ e_4\ \ldots\ e_n$. The rewriting procedure always terminates because it reduces the number of B. Thus, every B-term in $\mathbf{CL}(B)$ is equivalent to a B-term built by the syntax

$$e ::= B \mid B\ e \mid e \circ e \tag{1}$$

where $e_1 \circ e_2$ denotes $B\ e_1\ e_2$. We prefer to use the infix operator $\circ$ instead of B that has two arguments because associativity of B, that is, $B\ e_1\ (B\ e_2\ e_3) = B\ (B\ e_1\ e_2)\ e_3$ can be implicitly assumed. This simplifies the further discussion on B-terms. We will deal with only B-terms in syntax (1) from now on. The $\circ$ operator has a lower precedence than application in this paper, e.g., terms $B\ B \circ B$ and $B \circ B\ B$ represent $(B\ B) \circ B$ and $B \circ (B\ B)$, respectively.

The syntactic restriction by (1) does not suffice to proffer a canonical representation of B-terms. For example, both of the two B-terms $B \circ B\ B$ and $B\ (B\ B) \circ B$ are given in the form of (1), but we can see they are equivalent using (B3').

A *polynomial form of B-terms* is obtained by putting a restriction on the syntax so that no B combinator occurs outside of the $\circ$ operator while syntax (1) allows the B combinators and the $\circ$ operators to occur in an arbitrary position. The restricted syntax is given as

$$e ::= e_B \mid e \circ e \qquad e_B ::= B \mid B\ e_B$$

where terms in e_B have a form of $B(\dots(B(B\ B))\dots)$, that is $B^n B$ with some n, called *monomial*. The syntax can be simply rewritten into $e ::= B^n B \ \mid\ e \circ e$, which is called *polynomial*.

▶ **Definition 2.** A B-term $B^n B$ is called *monomial*. A *polynomial* is a B-term given in the form of

$$(B^{n_1} B) \circ (B^{n_2} B) \circ \cdots \circ (B^{n_k} B)$$

where $k > 0$ and $n_1, \dots, n_k \geq 0$ are integers. In particular, a polynomial is called *decreasing* when $n_1 \geq n_2 \geq \cdots \geq n_k$. The *length* of a polynomial P is a number of monomials in P, i.e., the length of the polynomial above is k. The numbers $n_1, n_2, \dots, n_k$ are called *degrees*.

In the rest of this subsection, we prove that for any B-term e there exists a unique decreasing polynomial equivalent to e. First, we show that e has an equivalent polynomial.

▶ **Lemma 3** ([12])**.** *For any B-term e, there exists a polynomial equivalent to e.*

Proof. We prove the statement by induction on the structure of e. In the case of $e \equiv B$, the term itself is polynomial. In the case of $e \equiv B\ e_1$, assume that e_1 has equivalent polynomial $(B^{n_1} B) \circ (B^{n_2} B) \circ \cdots \circ (B^{n_k} B)$. Repeatedly applying equation (B2') to $B\ e_1$, we obtain a polynomial equivalent to $B\ e_1$ as $(B^{n_1+1} B) \circ (B^{n_2+1} B) \circ \cdots \circ (B^{n_k+1} B)$. In the case of $e \equiv e_1 \circ e_2$, assume that e_1 and e_2 have equivalent polynomials P_1 and P_2, respectively. A polynomial equivalent to e is given by $P_1 \circ P_2$. ◀

Next, we show that for any polynomial P there exists a decreasing polynomial equivalent to P. A key equation of the proof is

$$(B^m B) \circ (B^n B) = (B^{n+1} B) \circ (B^m B) \quad \text{when } m < n, \tag{2}$$

which is shown by

$$\begin{aligned}
(B^m B) \circ (B^n B) &= B^m (B \circ (B^{n-m} B)) \\
&= B^m (B \circ (B\ (B^{n-m-1} B))) \\
&= B^m ((B(B(B^{n-m-1} B))) \circ B) \\
&= (B^{n+1} B) \circ (B^m B)
\end{aligned}$$

using equations (B2') and (B3').

▶ **Lemma 4.** *Any polynomial P has an equivalent decreasing polynomial P' such that*

- *the length of P and P' are equal, and*
- *the lowest degrees of P and P' are equal.*

Proof. We prove the statement by induction on the length of P. When the length is 1, that is, P is a monomial, P itself is decreasing and the statement holds. When the length k of P is greater than 1, take P_1 such that $P \equiv P_1 \circ (B^n B)$. From the induction hypothesis, there exists a decreasing polynomial $P_1' \equiv (B^{n_1} B) \circ (B^{n_2} B) \circ \cdots \circ (B^{n_{k-1}} B)$ equivalent to P_1, and the lowest degree of P_1 is n_{k-1}. If $n_{k-1} \geq n$, then $P' \equiv P_1' \circ (B^n\ B)$ is decreasing and equivalent to P. Since the lowest degrees of P and P' are n, the statement holds. If $n_{k-1} < n$, P is equivalent to

$$(B^{n_1}\ B) \circ \cdots \circ (B^{n_{k-1}} B) \circ (B^n B) = (B^{n_1} B) \circ \cdots \circ (B^{n+1} B) \circ (B^{n_{k-1}} B)$$

due to equation (2). Putting the last term as $P_2 \circ (B^{n_{k-1}} B)$, the length of P_2 is $k-1$ and the lowest degree of P_2 is greater than or equal to n_{k-1}. From the induction hypothesis, P_2 has an equivalent decreasing polynomial P'_2 of length $k-1$ and the lowest degree of P'_2 greater than or equal to n_{k-1}. Thereby we obtain a decreasing polynomial $P'_2 \circ (B^{n_{k-1}} B)$ equivalent to P and the statement holds. ◀

▶ **Example 5.** Consider a B-term $e = B\ (B\ B\ B)\ (B\ B)\ B$. First, applying equation (B1),

$$e = B\ (B\ B\ B)\ (B\ B)\ (B\ B) = B\ B\ B\ (B\ B\ (B\ B)) = B\ (B\ (B\ B\ (B\ B)))$$

so that every B has at most two arguments. Then replace each B to the infix $\circ$ operator if it has two arguments and obtain $B\ (B\ (B \circ (B\ B)))$ Applying equation (B2'), we have

$$\begin{aligned} B\ (B\ (B \circ (B\ B))) &= B\ ((B\ B) \circ (B\ (B\ B))) \\ &= (B\ (B\ B)) \circ (B\ (B\ (B\ B))) \\ &= (B^2 B) \circ (B^3 B). \end{aligned}$$

Applying equation (2), we obtain the decreasing polynomial $(B^4 B) \circ (B^2 B)$ equivalent to e.

Every B-term has at least one equivalent decreasing polynomial as shown so far. To conclude this subsection, we show the uniqueness of decreasing polynomial equivalent to any B-term, that is, every B-term e has no two distinct decreasing polynomials equivalent to e.

The proof is based on the idea that B-terms correspond to unlabeled binary trees. Let M be a term which is constructed from variables $x_1, \ldots, x_k$ and their applications. Then we can show that if the λ-term $\lambda x_1. \ldots \lambda x_k.\ M$ is in $\mathbf{CL}(B)$, then M is obtained by putting parentheses to some positions in the sequence $x_1\ \ldots\ x_k$. More precisely, we have the following lemma.

▶ **Lemma 6.** *Every λ-term in $\mathbf{CL}(B)$ is $\beta\eta$-equivalent to a λ-term of the form $\lambda x_1. \ldots \lambda x_k.\ M$ with some $k > 2$ where M satisfies the following two conditions: (1) M consists of only the variables $x_1, \ldots, x_k$ and their applications, and (2) for every subterm of M which is in the form of $M_1\ M_2$, if M_1 has a variable x_i, then M_2 does not have any variable x_j with $j \leq i$.*

Proof. By the structural induction of B-terms. ◀

From this lemma, we see that we do not need to specify variables in M and we can simply write like $\star\ \star\ (\star\ \star) =\ x_1\ x_2\ (x_3\ x_4)$. Formally speaking, every λ-term in $\mathbf{CL}(B)$ uniquely corresponds to a term built from $\star$ alone by the map $(\lambda x_1. \ldots \lambda x_k.\ M) \mapsto M[\star/x_1, \ldots, \star/x_k]$. We say an unlabeled binary tree (or simply, binary tree) for a term built from $\star$ alone since every term built from $\star$ alone can be seen as an unlabeled binary tree. (A term $\star$ corresponds to a leaf and $t_1\ t_2$ corresponds to the tree with left subtree t_1 and right subtree t_2.) To specify the applications in binary trees, we write $\langle t_1, t_2 \rangle$ for the application $t_1\ t_2$. For example, B-terms $B = \lambda x.\lambda y.\lambda z.\ x\ (y\ z)$ and $B\ B = \lambda x.\lambda y.\lambda z.\lambda w.\ x\ y\ (z\ w)$ are represented by $\langle \star, \langle \star, \star \rangle \rangle$ and $\langle \langle \star, \star \rangle, \langle \star, \star \rangle \rangle$, respectively.

We will present an algorithm for constructing the corresponding decreasing polynomial from a given binary tree. First let us define a function $\mathcal{L}_i$ with integer i which maps binary trees to lists of integers:

$$\mathcal{L}_i(\star) = [\ ] \qquad\qquad \mathcal{L}_i(\langle t_1, t_2 \rangle) = \mathcal{L}_{i+\|t_1\|}(t_2) +\!\!+ \mathcal{L}_i(t_1) +\!\!+ [i]$$

where $+\!\!+$ concatenates two lists and $\|t\|$ denotes a number of leaves. For example, $\mathcal{L}_0(\langle \langle \star, \star \rangle, \langle \star, \star \rangle \rangle) = [2, 0, 0]$ and $\mathcal{L}_1(\langle \langle \star, \langle \star, \star \rangle \rangle, \langle \star, \langle \star, \star \rangle \rangle \rangle) = [4, 4, 2, 1, 1]$. Informally, the

$\mathcal{L}_i$ function returns a list of integers which is obtained by labeling both leaves and nodes in the following steps. First each leaf of a given tree is labeled by $i, i+1, i+2, \ldots$ in left-to-right order. Then each binary node of the tree is labeled by the same label as its leftmost descendant leaf. The $\mathcal{L}_i$ functions return a list of only node labels in decreasing order. The length of the list equals the number of nodes, that is, smaller by one than the number of variables in the λ-term.

We define a function $\mathcal{L}$ which takes a binary tree t and returns a list of non-negative integers in $\mathcal{L}_{-1}(t)$, that is, the list obtained by excluding trailing all -1's in $\mathcal{L}_{-1}(t)$. Note that by excluding the label -1's it may happen to be $\mathcal{L}(t) = \mathcal{L}(t')$ for two distinct binary trees t and t' even though the $\mathcal{L}_i$ function is injective. However, those binary trees t and t' must be 'η-equivalent' in terms of the corresponding λ-terms.

The following lemma claims that the $\mathcal{L}$ function computes a list of degrees of a decreasing polynomial corresponding to a given λ-term.

▶ **Lemma 7.** *A decreasing polynomial $(B^{n_1}B) \circ (B^{n_2}B) \circ \cdots \circ (B^{n_k}B)$ is $\beta\eta$-equivalent to a λ-term $e \in \mathbf{CL}(B)$ corresponding a binary tree t such that $\mathcal{L}(t) = [n_1, n_2, \ldots, n_k]$.*

Proof. We prove the statement by induction on the length of the polynomial P.

When $P \equiv B^n B$ with $n \geq 0$, it is found to be equivalent to the λ-term

$$\lambda x_1.\lambda x_2.\lambda x_3.\ldots.\lambda x_{n+1}.\lambda x_{n+2}.\lambda x_{n+3}.\ x_1\ x_2\ x_3\ \ldots\ x_{n+1}\ (x_{n+2}\ x_{n+3})$$

by induction on n. This λ-term corresponds to a binary tree $t = \langle\langle\ldots\langle\langle\star, \underbrace{\star\rangle, \star\rangle, \ldots, \star\rangle}_{n \text{ leaves}}, \langle\star, \star\rangle\rangle$.

Then we have $\mathcal{L}(t) = [n]$ holds from $\mathcal{L}_{-1}(t) = [n, \underbrace{-1, -1, \ldots, -1}_{n+1}]$.

When $P \equiv P' \circ (B^n B)$ with $P' \equiv (B^{n_1}B) \circ \cdots \circ (B^{n_k}B)$, $k \geq 1$ and $n_1 \geq \cdots \geq n_k \geq n \geq 0$, there exists a λ-term $\beta\eta$-equivalent to P' corresponding a binary tree t' such that $\mathcal{L}(t') = [n_1, \ldots, n_k]$ from the induction hypothesis. The binary tree t' must have the form of $\langle\langle\langle\ldots\langle\langle\star, \underbrace{\star\rangle, \star\rangle, \ldots, \star\rangle}_{n_k \text{ leaves}}, t_1\rangle, \ldots, t_m\rangle$ with $m \geq 1$ and some trees $t_1, \ldots, t_m$, otherwise $\mathcal{L}(t')$ would contain an integer smaller than n_k. From the definition of $\mathcal{L}$ and $\mathcal{L}_i$, we have

$$\mathcal{L}(t') = \mathcal{L}_{s_m}(t_m) + \!\!\!+ \cdots + \!\!\!+ \mathcal{L}_{s_1}(t_1) \tag{3}$$

where $s_j = n_k + 1 + \sum_{i=1}^{j-1} \|t_i\|$. Additionally, the structure of t' implies $P' = \lambda x_1.\ldots.\lambda x_l.\ x_1\ x_2\ \ldots\ x_{n_k+1}\ e_1 \ldots e_m$ where e_i corresponds to a binary tree t_i for $i = 1, \ldots, m$. From $B^n\ B = \lambda y_1.\ldots.\lambda y_{n+3}.\ y_1\ y_2 \ldots y_{n+1}\ (y_{n+2}\ y_{n+3})$, we compute a λ-term $\beta\eta$-equivalent to $P \equiv P' \circ (B^n B)$ by

$$\begin{aligned}
P &= \lambda x.\ P'(B^n B\ x) \\
&= \lambda x.\ (\lambda x_1.\ldots.\lambda x_l.\ x_1\ x_2 \ldots x_{n_k+1}\ e_1\ \ldots\ e_m) \\
&\qquad\quad (\lambda y_2.\ldots.\lambda y_{n+3}.\ x\ y_2 \ldots y_{n+1}\ (y_{n+2}\ y_{n+3})) \\
&= \lambda x.\lambda x_2.\ldots.\lambda x_l.\ (\lambda y_2.\ldots.\lambda y_{n+3}.\ x\ y_2 \ldots y_{n+1}\ (y_{n+2}\ y_{n+3}))\ x_2 \ldots x_{n_k+1}\ e_1 \ldots e_m \\
&= \lambda x.\lambda x_2.\ldots.\lambda x_l. \\
&\qquad (\lambda y_{n+1}.\lambda y_{n+2}.\lambda y_{n+3}.\ x\ x_2 \ldots x_n\ y_{n+1}\ (y_{n+2}\ y_{n+3}))\ x_{n+1} \ldots x_{n_k+1}\ e_1 \ldots e_m
\end{aligned}$$

where $n_k \geq n$ is taken into account. We split into four cases: (i) $n_k = n$ and $m = 1$, (ii) $n_k = n$ and $m > 1$, (iii) $n_k = n + 1$, and (iv) $n_k > n + 1$. In the case (i) where $n_k = n$ and $m = 1$, we have

$$P = \lambda x.\lambda x_2.\ldots.\lambda x_l.\lambda y_{n+3}.\ x\ x_2 \ldots x_n\ x_{n+1}\ (e_1\ y_{n+3}).$$

whose corresponding binary tree t is $\langle\langle\underbrace{\ldots\langle\langle\star,\star\rangle,\star\rangle,\ldots,\star\rangle}_{n \text{ leaves}},\langle t_1,\star\rangle\rangle$. From equation (3), $\mathcal{L}(t) = \mathcal{L}_{n+1}(t_1) +\!\!+ [n+1] = \mathcal{L}(t') +\!\!+ [n+1] = [n_1,\ldots,n_k,n+1]$, thus the statement holds. In the case (ii) where $n_k = n$ and $m > 1$, we have

$$P = \lambda x.\lambda x_2.\ldots.\lambda x_l.\ x\ x_2 \ldots x_n\ x_{n+1}\ (e_1\ e_2)\ e_3 \ldots e_m.$$

whose corresponding binary tree t is $\langle\langle\langle\underbrace{\ldots\langle\langle\star,\star\rangle,\star\rangle,\ldots,\star\rangle}_{n \text{ leaves}},\langle t_1,t_2\rangle,t_3\rangle,\ldots,t_m\rangle$. Hence, $\mathcal{L}(t) = \mathcal{L}(t') +\!\!+ [n+1]$ holds again from equation (3). In the case (iii) where $n_k = n+1$, we have

$$P = \lambda x.\lambda x_2.\ldots.\lambda x_l.\ x\ x_2 \ldots x_n\ x_{n+1}\ (x_{n+2}\ e_1)\ e_2 \ldots e_m, \text{ or}$$

whose corresponding binary tree t is $\langle\langle\langle\underbrace{\ldots\langle\langle\star,\star\rangle,\star\rangle,\ldots,\star\rangle}_{n \text{ leaves}},\langle\star,t_1\rangle,t_2\rangle,\ldots,t_m\rangle$. Hence, $\mathcal{L}(t) = \mathcal{L}(t') +\!\!+ [n+1]$ holds from equation (3). In the case (iv) where $n_k \geq n+2$, we have

$$P = \lambda x.\lambda x_2.\ldots.\lambda x_l.\ x\ x_2 \ldots x_n\ x_{n+1}\ (x_{n+2}\ x_{n+3})\ \ldots\ e_1 \ldots e_m,$$

whose corresponding binary tree t is $\langle\langle\langle\underbrace{\ldots\langle\langle\star,\star\rangle,\star\rangle,\ldots,\star\rangle}_{n \text{ leaves}},\langle\star,\star\rangle,\ldots,t_1\rangle,\ldots,t_m\rangle$. Hence, $\mathcal{L}(t) = \mathcal{L}(t') +\!\!+ [n+1]$ holds from equation (3). ◀

▶ **Example 8.** A λ-term $\lambda x_1.\lambda x_2.\lambda x_3.\lambda x_4.\lambda x_5.\lambda x_6.\lambda x_7.\lambda x_8.\ x_1\ (x_2\ x_3)\ (x_4\ x_5\ x_6\ (x_7\ x_8))$ is $\beta\eta$-equivalent to $(B^5\ B) \circ (B^2\ B) \circ (B^2\ B) \circ (B^2\ B) \circ (B^0\ B)$ because its corresponding binary tree $t = \langle\langle\star,\langle\star,\star\rangle\rangle,\langle\langle\langle\star,\star\rangle,\star\rangle,\langle\star,\star\rangle\rangle\rangle$ satisfies $\mathcal{L}(t) = [5,2,2,2,0]$.

The previous lemmas immediately conclude the uniqueness of decreasing polynomials for B-terms shown in the following theorem.

▶ **Theorem 9.** *Every B-term e has a unique decreasing polynomial.*

Proof. For any given B-term e, we can find a decreasing polynomial for e from Lemma 3 and Lemma 4. Since every decreasing polynomial corresponds to only one binary tree (and since every B-term also corresponds to only one binary tree up to η-equivalence) from Lemma 7, the present statement holds. ◀

This theorem implies that the decreasing polynomial of B-terms can be used as their *canonical representation*, which is effectively derived as shown in Lemma 3 and Lemma 4.

As a corollary of the theorem, we can show the "only if" statement of Theorem 1, which corresponds to the completeness of the equation system.

Proof. Let e_1 and e_2 be equivalent B-terms, that is, their λ-terms are $\beta\eta$-equivalent. From Theorem 9, their decreasing polynomials are the same. Since the decreasing polynomial is derived from e_1 and e_2 by equations (B1), (B2), and (B3) according to the proofs of Lemma 3 and Lemma 4, equivalence between e_1 and e_2 is also derived from these equations. ◀

4 Results on the ρ-property of B-terms

We investigate the ρ-property of concrete B-terms, some of which have the property and others do not. For B-terms having the ρ-property, we introduce an efficient implementation to compute the entry point and the size of the cycle. For B-terms not having the ρ-property, we give a proof why they do not have.

4.1 B-terms having the ρ-property

As shown in Section 2, we can check that B-terms equivalent to $B^n B$ with $n \leq 6$ have the ρ-property by computing $(B^n B)_{(i)}$ for each i. However, it is not easy to check it by computer without an efficient implementation because we should compute all $(B^6 B)_{(i)}$ with $i \leq 2980054085040$ $(= 2641033883877 + 339020201163)$ to know that $\rho(B^6 B) = (2641033883877, 339020201163)$. A naive implementation which computes terms of $(B^6 B)_{(i)}$ for all i and stores all of them has no hope to detect the ρ-property.

We introduce an efficient procedure to find the ρ-property of B-terms which can successfully compute $\rho(B^6 B)$. The procedure is based on two orthogonal ideas, Floyd's cycle-finding algorithm [6] and an efficient right application algorithm over decreasing polynomials presented in Section 3.3.

The first idea, Floyd's cycle-finding algorithm (also called the tortoise and the hare algorithm), enables us to detect the cycle with a constant memory usage, that is, the history of all terms $X_{(i)}$ does not need to be stored to check the ρ-property of the X combinator. The key of this algorithm is the fact that there are two distinct integers i and j with $X_{(i)} = X_{(j)}$ if and only if there is an integer m with $X_{(m)} = X_{(2m)}$, where the latter requires to compare $X_{(i)}$ and $X_{(2i)}$ from smaller i and store only these two terms for the next comparison between $X_{(i+1)} = X_{(i)}X$ and $X_{(2i+2)} = X_{(2i)}XX$ when $X_{(i)} \neq X_{(2i)}$. The following procedure computes the entry point and the size of the cycle if X has the ρ-property.

1. Find the smallest m such that $X_{(m)} = X_{(2m)}$.
2. Find the smallest k such that $X_{(k)} = X_{(m+k)}$.
3. Find the smallest $0 < c \leq k$ such that $X_{(m)} = X_{(m+c)}$. If not found, put $c = m$.

After this procedure, we find $\rho(X) = (k, c)$. The third step can be run in parallel during the second one. See [6, exercise 3.1.6] for the detail. One could use slightly more (possibly) efficient algorithms by Brent [3] and Gosper [2, item 132] for cycle detection.

Efficient cycle-finding algorithms do not suffice to compute $\rho(B^6 B)$. Only with the idea above running on a laptop (1.7 GHz Intel Core i7 / 8GB of memory), it takes about 2 hours even for $\rho(B^5 B)$ and fails to compute $\rho(B^6 B)$ with an out-of-memory error.

The second idea enables us to efficiently compute $X_{(i+1)}$ from $X_{(i)}$ for B-terms X. The key of this algorithm is to use the canonical representation of $X_{(i)}$, that is a decreasing polynomial, and directly compute the canonical representation of $X_{(i+1)}$ from that of $X_{(i)}$. Additionally, the canonical representation enables us to quickly decide equivalence which is required many times to find the cycle. It takes time just proportional to their lengths. If λ-terms are used for finding the cycle, both application and deciding equivalence require much more complicated computation. Our implementation based on these two ideas computes $\rho(B^5 B)$ and $\rho(B^6 B)$ in 10 minutes and 59 days (!), respectively.

For two given decreasing polynomials P_1 and P_2, we show how a decreasing polynomial P equivalent to $(P_1\ P_2)$ can be obtained. The method is based on the following lemma about application of one B-term to another B-term.

▶ **Lemma 10.** *For B-terms e_1 and e_2, there exists $k \geq 0$ such that $e_1 \circ (B\ e_2) = B\ (e_1\ e_2) \circ B^k$.*

Proof. Let P_1 be a decreasing polynomial equivalent to e_1. We prove the statement by case analysis on the maximum degree in P_1. When the maximum degree is 0, we can take $k' \geq 1$ such that $P_1 \equiv \underbrace{B \circ \cdots \circ B}_{k'} = B^{k'}$. Then,

$$e_1 \circ (B\ e_2) = \underbrace{B \circ \cdots \circ B}_{k'} \circ (B\ e_2) = (B^{k'+1} e_2) \circ \underbrace{B \circ \cdots \circ B}_{k'} = B\ (e_1\ e_2) \circ B^{k'}$$

where equation (B3') is used k' times in the second equation. Therefore the statement holds by taking $k = k'$. When the maximum degree is greater than 0, we can take a decreasing polynomial P' for a B-term and $k' \geq 0$ such that $P_1 = (B\ P') \circ \underbrace{B \circ \cdots \circ B}_{k'} = (B\ P') \circ B^{k'}$ due to equation (B2'). Then,

$$
\begin{aligned}
e_1 \circ (B\ e_2) &= (B\ P') \circ \underbrace{B \circ \cdots \circ B}_{k'} \circ (B\ e_2) \\
&= (B\ P') \circ (B^{k'+1} e_2) \circ \underbrace{B \circ \cdots \circ B}_{k'} \\
&= B\ (P' \circ (B^{k'} e_2)) \circ B^{k'} \\
&= B\ (B\ P'\ (B^{k'} e_2)) \circ B^{k'} \\
&= B\ (P_1\ e_2) \circ B^{k'} \\
&= B\ (e_1\ e_2) \circ B^{k'}.
\end{aligned}
$$

Therefore, the statement holds by taking $k = k'$. ◀

This lemma indicates that, from two decreasing polynomials P_1 and P_2, a decreasing polynomial P equivalent to $(P_1\ P_2)$ can be obtained in the following steps where L_1 and L_2 are lists of non-negative numbers as shown in Section 3.3 corresponding to P_1 and P_2.

1. Build P_2' by incrementing each degree of P_2 by 1, i.e., when $P_2 \equiv (B^{n_1} B) \circ \cdots \circ (B^{n_l} B)$, $P_2' \equiv (B^{n_1+1} B) \circ \cdots \circ (B^{n_l+1} B)$. In terms of the list representation, a list L_2' is built from L_2 by incrementing each value by 1.
2. Find a decreasing polynomial P_{12} corresponding to $P_1 \circ P_2'$ by equation (2). In terms of the list representation, a list L_{12} is constructed by appending L_1 and L_2' and repeatedly applying (2).
3. Obtain P by decrementing each degree of P_{12} after eliminating the trailing 0-degree units, i.e., when $P_{12} \equiv (B^{n_1} B) \circ \cdots \circ (B^{n_l} B) \circ (B^0 B) \circ \cdots \circ (B^0 B)$ with $n_1 \geq \cdots \geq n_l > 0$, $P \equiv (B^{n_1-1} B) \circ \cdots \circ (B^{n_l-1} B)$. In terms of the list representation, a list L is obtained from L_{12} by decrementing each value by 1 after removing trailing 0's.

In the first step, a decreasing polynomial P_2' equivalent to $B\ P_2$ is obtained. The second step yields a decreasing polynomial P_{12} for $P_1 \circ P_2' = P_1 \circ (B\ P_2)$. Since P_1 and P_2 are decreasing, it is easy to find P_{12} by repetitive application of equation (2) for each unit of P_2', à la insertion operation in insertion sort. In the final step, a polynomial P that satisfies $(B\ P) \circ B^k = P_{12}$ with some k is obtained. From Lemma 10 and the uniqueness of decreasing polynomials, P is equivalent to $(P_1\ P_2)$.

▶ **Example 11.** Let P_1 and P_2 be decreasing polynomials represented by lists $L_1 = [4, 1, 0]$ and $L_2 = [2, 0]$. Then a decreasing polynomial P equivalent to $(P_1\ P_2)$ is obtained as a list L in three steps:

1. A list $L_2' = [3, 1]$ is obtained from L_2 by incrementing each value by 1.
2. A decreasing list L_{12} is obtained from L_1 and L_2' by

   $$L_{12} = [4, 1, \underline{0, 3}, 1] = [4, \underline{1, 4}, 0, 1] = [\underline{4, 5}, 1, 0, 1] = [6, 4, 1, \underline{0, 1}] = [6, 4, \underline{1, 2}, 0] = [6, 4, 3, 1, 0]$$

   where equation (2) is applied in each underlined pair.
3. A list $L = [5, 3, 2, 0]$ is obtained from L_{12} as the result of the application by decrementing each value by 1 after removing trailing 0's.

The implementation based on the right application over decreasing polynomials is available at `https://github.com/ksk/Rho`. Note that the program does not terminate for the combinator which does not have the ρ-property. It will not help to decide if a combinator has the ρ-property. One might observe how the terms grow by repetitive right applications through running the program, though.

4.2 B-terms not having the ρ-property

We prove that the B-terms $(B^k B)^{(k+2)n}$ $(k \geq 0,\ n > 0)$ do not have the ρ-property. For example, B-term $B^2 = B\ B\ B$, which is the case of $k = 0$ and $n = 1$, does not have the ρ-property. To this end, we show that the number of variables in the $\beta\eta$-normal form of $((B^k B)^{(k+2)n})_{(i)}$ is monotonically non-decreasing and that it implies the anti-ρ-property. Additionally, after proving that, we consider a sufficient condition not to have the ρ-property through the monotonicity.

First, we introduce some notations. Suppose that the $\beta\eta$-normal form of a B-term X is given by $\lambda x_1. \dots \lambda x_n.\ x_1\ e_1\ \cdots\ e_k$ for some terms $e_1, \dots, e_k$. Then we define $l(X) = n$ (the number of variables), $a(X) = k$ (the number of arguments of x_1), and $N_i(X) = e_i$ for $i = 1, \dots, k$. For convinience, we define functions l, a, and N_i also for terms of form $Y = x\ e_1 \dots e_k$ in the same mannar. That is, $l(Y)$ is the number of variables in Y, $a(Y) = k$, and $N_i(Y) = e_i$. Let X' be another B-term and suppose its $\beta\eta$-normal form is given by $\lambda x'_1. \dots \lambda x'_{n'}.\ e'$ where e' does not have λ-abstractions. We can see $X\ X' = (\lambda x_1. \dots \lambda x_n.\ x_1\ e_1\ \cdots\ e_k)\ X' = \lambda x_2. \dots \lambda x_n.\ X'\ e_1\ \cdots\ e_k$ and from Lemma 6, its $\beta\eta$-normal form is

$$\begin{cases} \lambda x_2. \dots \lambda x_n. \lambda x'_{k+1}. \dots \lambda x'_{n'}.\ e'[e_1/x'_1, \dots, e_k/x'_k] & (k \leq n') \\ \lambda x_2. \dots \lambda x_n.\ e'[e_1/x'_1, \dots, e_{n'}/x'_{n'}]\ e_{n'+1}\ \cdots\ e_k & \text{(otherwise).} \end{cases}$$

Here $e'[e_1/x'_1, \dots, e_k/x'_k]$ is the term which is obtained by substituting $e_1, \dots, e_k$ to the variables $x'_1, \dots, x'_k$ in e' .

By simple computation with this fact, we get the following lemma:

▶ **Lemma 12.** *Let X and X' be B-terms. Then*

$$\begin{aligned} l(X\ X') &= l(X) - 1 + \max\{l(X') - a(X),\ 0\} \\ a(X\ X') &= a(X') + a(N_1(X)) + \max\{a(X) - l(X'),\ 0\} \\ N_1(X\ X') &= \begin{cases} N_1(X')[N_2(X)/x'_2, \dots, N_m(X)/x'_m] & (\textit{if } N_1(X) \textit{ is a variable}) \\ N_1(N_1(X)) & (\textit{otherwise}) \end{cases} \end{aligned}$$

where $m = \min\{l(N_1(X')),\ a(X)\}$.

The $\beta\eta$-normal form of $(B^k B)^{(k+2)n}$ is given by

$$\lambda x_1. \dots \lambda x_{k+(k+2)n+2}.\ x_1\ x_2\ \cdots\ x_{k+1}\ (x_{k+2}\ x_{k+3}\ \cdots\ x_{k+(k+2)n+2}).$$

This is deduced from Lemma 7 since the binary tree corresponding to the above λ-term is $t = \langle \langle \dots \langle \underbrace{\langle \star, \star \rangle, \star \rangle, \dots, \star}_{k+1} \rangle, \langle \dots \langle \langle \star, \underbrace{\star \rangle, \star \rangle, \dots, \star}_{(k+2)n} \rangle \rangle$ and $\mathcal{L}(t) = [\underbrace{k, \dots, k}_{(k+2)n}]$. Especially, we get $l((B^k B)^{(k+2)n}) = k + (k+2)n + 2$. In this section, we write $\langle \star, \star, \star, \dots, \star \rangle$ for $\langle \dots \langle \langle \star, \star \rangle, \star \rangle, \dots, \star \rangle$ and identify B-terms with their corresponding binary trees.

To describe properties of $(B^k B)^{(k+2)n}$, we introduce a set $T_{k,n}$ which is closed under right application of $(B^k B)^{(k+2)n}$, that is, $T_{k,n}$ satisfies that "if $X \in T_{k,n}$ then $X\ (B^k B)^{(k+2)n} \in T_{k,n}$ holds". First we inductively define a set of terms $T'_{k,n}$ as follows:

1. $\star \in T'_{k,n}$
2. $\langle \star,\ s_1,\ \ldots,\ s_{(k+2)n}\rangle \in T'_{k,n}$ if $s_i = \star$ for each multiple i of $k+2$ and $s_i \in T'_{k,n}$ for the others.

Then we define $T_{k,n}$ by $T_{k,n} = \left\{ \langle t_0,\ t_1,\ \ldots,\ t_{k+1}\rangle \;\middle|\; t_0,\ t_1, \ldots, t_{k+1} \in T'_{k,n} \right\}$. It is obvious that $(B^k B)^{(k+2)n} \in T_{k,n}$. Now we shall prove that $T_{k,n}$ is closed under right application of $(B^k B)^{(k+2)n}$.

▶ **Lemma 13.** *If $X \in T_{k,n}$ then $X\ (B^k B)^{(k+2)n} \in T_{k,n}$.*

Proof. From the definition of $T_{k,n}$, if $X \in T_{k,n}$ then X can be written in the form $\langle t_0,\ t_1,\ \ldots,\ t_{k+1}\rangle$ for some $t_0, \ldots, t_{k+1} \in T'_{k,n}$. In the case where $t_0 = \star$, we have $X\ (B^k B)^{(k+2)n} = \langle t_1,\ \ldots,\ t_{k+1},\ \langle \underbrace{\star,\ \ldots,\ \star}_{(k+2)n} \rangle\rangle \in T_{k,n}$. In the case where t_0 has the form of 2 in the definition of $T'_{k,n}$, then we have $X = \langle \star,\ s_1,\ \ldots,\ s_{(k+2)n},\ t_1,\ \ldots,\ t_{k+1}\rangle$ with $s_i = \star$ for each multiple i of $k+2$ and $s_i \in T'_{k,n}$ for the others, hence

$$X\ (B^k B)^{(k+2)n} = \langle s_1,\ \ldots,\ s_{k+1},\ \langle s_{k+2},\ \ldots,\ s_{(k+2)n},\ t_1,\ \ldots,\ t_{k+1},\ \star\rangle\rangle.$$

We can easily see $s_1,\ \ldots,\ s_{k+1}$, and $\langle s_{k+2},\ \ldots,\ s_{(k+2)n},\ t_1,\ \ldots,\ t_{k+1},\ \star\rangle$ are in $T'_{k,n}$. ◀

From the definition of $T_{k,n}$, we can compute that $a(X)$ equals $k+1$ or $(k+2)n+k+1$ if $X \in T_{k,n}$. Particularly, we get the following:

▶ **Lemma 14.** *For any $X \in T_{k,n}$, $a(X) \le (k+2)n + k + 1 = l((B^k B)^{(k+2)n}) - 1$.*

This lemma is crucial to show that the number of variables in $((B^k B)^{(k+2)n})_{(i)}$ is monotonically non-decreasing. Put $Z = (B^k B)^{(k+2)n}$ for short. Since $Z \in T_{k,n}$, we have $\{Z_{(i)} \mid i \ge 1\} \subset T_{k,n}$ by Lemma 13. Using Lemma 14, we can simplify Lemma 12 in the case where $X = Z_{(i)}$ and $X' = Z$ as follows:

$$l(Z_{(i+1)}) = l(Z_{(i)}) + (k+2)n + k + 1 - a(Z_{(i)}) \tag{4}$$

$$a(Z_{(i+1)}) = a(N_1(Z_{(i)})) + k + 1 \tag{5}$$

$$N_1(Z_{(i+1)}) = \begin{cases} N_2(Z_{(i)}) & \text{(if } N_1(Z_{(i)}) \text{ is a variable)} \\ N_1(N_1(Z_{(i)})) & \text{(otherwise).} \end{cases} \tag{6}$$

By (4) and Lemma 14, we get $l(Z_{(i+1)}) \ge l(Z_{(i)})$.

To prove that Z does not have the ρ-property, it suffices to show the following:

▶ **Lemma 15.** *For any $i \ge 1$, there exists $j > i$ that satisfies $l(Z_{(j)}) > l(Z_{(i)})$.*

Proof. Suppose that there exists $i \ge 1$ that satisfies $l(Z_{(i)}) = l(Z_{(j)})$ for any $j > i$. We get $a(Z_{(j)}) = (k+2)n + k + 1$ by (4) and then $a(N_1(Z_{(j)})) = (k+2)n$ by (5). Therefore $N_1(Z_{(j)})$ is not a variable for any $j > i$ and from (6), we obtain $N_1(Z_{(j)}) = N_1(N_1(Z_{(j-1)})) = \cdots = \underbrace{N_1(\cdots N_1(}_{j-i+1} Z_{(i)}) \cdots)$ for any $j > i$. However, this implies that $Z_{(i)}$ has infinitely many variables and it yields contradiction. ◀

Now, we get the desired result:

▶ **Theorem 16.** *For any $k \ge 0$ and $n > 0$, $(B^k B)^{(k+2)n}$ does not have the ρ-property.*

The key fact which enables us to show the anti-ρ-property of $(B^k B)^{(k+2)n}$ is the existence of the set $T_{k,n} \supset \left\{ ((B^k B)^{(k+2)n})_{(i)} \;\middle|\; i \ge 1 \right\}$ which satisfies Lemma 14. In a similar way, we can show the anti-ρ-property of a B-term which has such a "good" set. That is,

▶ **Theorem 17.** *Let X be a B-term and T be a set of B-terms. If $\left\{X_{(i)} \;\middle|\; i \geq 1\right\} \subset T$ and $l(X) \geq a(X') + 1$ for any $X' \in T$, then X does not have the ρ-property.*

Here is an example of the B-terms which satisfy the condition in Theorem 17 with some set T. Consider $X = (B^2B)^2 \circ (BB)^2 \circ B^2 = \langle\star,\ \langle\star,\ \langle\star,\ \langle\star,\ \star,\ \star\rangle,\ \star\rangle,\ \star\rangle\rangle$. We inductively define T' as follows:

1. $\star \in T'$
2. For any $t \in T'$, $\langle\star,\ t,\ \star\rangle \in T'$
3. For any $t_1, t_2 \in T'$, $\langle\star,\ t_1,\ \star,\ \langle\star,\ t_2,\ \star\rangle,\ \star\rangle \in T'$

Then $T = \{\langle t_1,\ \langle\star,\ t_2,\ \star\rangle\rangle \mid t_1, t_2 \in T'\}$ satisfies the condition in Theorem 17. It can be checked simply by case analysis. Thus

▶ **Theorem 18.** *$(B^2B)^2 \circ (BB)^2 \circ B^2$ does not have the ρ-property.*

Theorem 17 gives a possible technique to prove the monotonicity with respect to $l(X_{(i)})$, or, the anti-ρ-property of X, for some B-term X. Moreover, we can consider another problem on B-terms: "Give a necessary and sufficient condition to have the monotonicity for B-terms."

5 Concluding remark

We have investigated the ρ-properties of B-terms in particular forms so far. While the B-terms equivalent to B^nB with $n \leq 6$ have the ρ-property, the B-terms $(B^kB)^{(k+2)n}$ with $k \geq 0$ and $n > 0$ and $(B^2B)^2 \circ (BB)^2 \circ B^2$ do not. In this section, remaining problems related to these results are introduced and possible approaches to illustrate them are discussed.

5.1 Remaining problems

The ρ-property is defined for any combinatory terms (and closed λ-terms). We investigated it only for B-terms as a simple but interesting instance in the present paper. From his observation on repetitive right applications for several B-terms, Nakano [8] has conjectured as follows.

▶ **Conjecture 19.** *A B-term e has the ρ-property if and only if e is a monomial, i.e., e is equivalent to B^nB with $n \geq 0$.*

The "if" part for $n \leq 6$ has been shown by computation and the "only if" part for $(B^kB)^{(k+2)n}$ $(k \geq 0, n > 0)$ and $(B^2B)^2 \circ (BB)^2 \circ B^2$ has been shown by Theorem 16. This conjecture implies that the ρ-property of B-terms is decidable. We conjecture that the ρ-property of even BCK- and BCI-terms is decidable. The decidability for the ρ-property of S-terms and L-terms can also be considered. Waldmann's work on a rational representation of normalizable S-terms may be helpful to solve it. We expect that none of S-terms have the ρ-property as S itself does not, though. Regarding L-terms, Statman's work [11] may be helpful where equivalence of L-terms is shown decidable up to a congruence relation induced by $L\ e_1\ e_2 \to e_1\ (e_2\ e_2)$. It would be interesting to investigate the ρ-property of L-terms in this setting.

5.2 Possible approaches

The present paper introduces a canonical representation to make equivalence check of B-terms easier. The idea of the representation is based on that we can lift all $\circ$'s (2-argument B) to the outside of B (1-argument B) by equation (B2'). One may consider it the other way around. Using the equation, we can lift all B's (1-argument B) to the outside of $\circ$ (2-argument B).

Then one of the arguments of $\circ$ becomes B. By equation (B3'), we can move all B's right. Thereby we find another canonical representation for B-terms given by

$$e ::= B \mid B\ e \mid e \circ B$$

whose uniqueness could be easily proved in a way similar to Theorem 9.

Waldmann [13] suggests that the ρ-property of $B^n B$ may be checked even without converting B-terms into canonical forms. He simply defines B-terms by

$$e ::= B^k \mid e\ e$$

and regards B^k as a constant which has a rewrite rule $B^k\ e_1\ e_2\ \ldots\ e_{k+2} \to e_1\ (e_2\ \ldots\ e_{k+2})$. He implemented a check program in Haskell to confirm the ρ-property. Even in the restriction on rewriting rules, he found that $(B^0B)_{(9)} = (B^0B)_{(13)}$, $(B^1B)_{(36)} = (B^1B)_{(56)}$, $(B^2B)_{(274)} = (B^2B)_{(310)}$ and $(B^3B)_{(4267)} = (B^3B)_{(10063)}$, in which it requires a few more right applications to find the ρ-property than the case of canonical representation. If the ρ-property of B^nB for any $n \geq 0$ is shown under the restricted equivalence given by rewriting rules, then we can conclude the "if" part of Conjecture 19.

Another possible approach is to observe the change of (principal) types by right repetitive application. Although there are many distinct λ-terms of the same type, we can consider a desirable subset of typed λ-terms. As shown by Hirokawa [5], each BCK-term can be characterized by its type, that is, any two λ-terms in $\mathbf{CL}(BCK)$ of the same principal type are identical up to β-equivalence. This approach may require observing unification between types in a clever way.

References

1 Hendrik P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics*. Studies in Logic and the Foundations of Mathematics. Elsevier Science, 1984.

2 Michael Beeler, Ralph W. Gosper, and Richard C. Schroeppel. HAKMEM. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1972.

3 Richard P. Brent. An improved Monte Carlo factorization algorithm. *BIT*, 20(2):176–184, 1980.

4 Haskell B. Curry. Grundlagen der Kombinatorischen Logik (Teil II). *American Journal of Mathematics*, 52(4):789–834, 1930.

5 Sachio Hirokawa. Principal types of BCK-lambda-terms. *Theoretical Computer Science*, 107(2):253–276, Jan 1993.

6 Donald E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1997.

7 Evgeny S. Ljapin. *Semigroups*. Translations of Mathematical Monographs. American Mathematical Society, 1968.

8 Keisuke Nakano. ρ-property of combinators. 29th TRS Meeting in Tokyo, 2008. URL: `http://www.jaist.ac.jp/~hirokawa/trs-meeting/original/29.html`.

9 Chris Okasaki. Flattening combinators: surviving without parentheses. *Journal of Functional Programming*, 13(4):815–822, July 2003.

10 Raymond M. Smullyan. *To Mock a Mockingbird*. Knopf Doubleday Publishing Group, 2012.

11 Rick Statman. The Word Problem for Smullyan's Lark Combinator is Decidable. *Journal of Symbolic Computation*, 7(2):103–112, 1989.

12 Rick Statman. To Type A Mockingbird. Draft paper available from `http://tlca.di.unito.it/PAPER/TypeMock.pdf`, December 2011.

13 Johannes Waldmann. Personal communication, March 2013.