A term rewriting characterization of the polytime functions and related complexity classes

Arnold Beckmann and Andreas Weiermann
Institut für mathematische Logik und Grundlagenforschung
der Westfälischen Wilhelms-Universität Münster
Einsteinstr. 62
D-48149 Münster
Germany

February 23, 2004

Abstract

A natural term rewriting framework for the Bellantoni Cook schemata of predicative recursion, which yields a canonical definition of the polynomial time computable functions, is introduced. In terms of an exponential function both, an upper bound and a lower bound are proved for the resulting derivation lengths of the functions in question. It is proved that any natural reduction strategy yields an algorithm which runs in exponential time. We give an example in which this estimate is tight. It is proved that the resulting derivation lengths become polynomially bounded in the lengths of the inputs if the rewrite rules are only applied to terms in which the safe arguments – no restrictions are assumed for the normal arguments – consist of values, i.e. numerals, and not of names, i.e. non numeral terms. It is proved that in the latter situation any inside first reduction strategy and any head reduction strategy yield algorithms, for the function under consideration, for which the running time is bounded by an appropriate polynomial in the lengths of the input. A feasible rewrite system for predicative recursion with predicative parameter substitution is defined. It is proved that the derivation lengths of this rewrite system are polynomially bounded in the lengths of the inputs. As a corollary we reobtain Bellantoni's result stating that predicative recursion is closed under predicative parameter recursion.

1 Introduction

This article is part of a general investigation on term rewriting applications to (sub-) recursive function theory. The underlying idea of this program – which has been initiated in Cichon and Weiermann 1995 [6] – is as follows. Fix an inductively defined complexity class $\mathcal{C}$ of recursive functions. Assume that each function $\mathcal{F} \in \mathcal{C}$ is defined via an equation (or a system of equations) of the following form

$$\mathcal{F}(\vec{x}) = \mathcal{T}(\lambda\vec{y}.\mathcal{F}(\vec{y}), \vec{x})$$

where $\mathcal{T}$ involves some previously defined functions from $\mathcal{C}$.

The corresponding rewrite system $R_{\mathcal{C}}$ is defined as follows. The signature $\mathcal{S}IG(\mathcal{C})$ consists of function symbols for the functions in $\mathcal{C}$. It is assumed that there is a map Φ which assigns to each $f \in \mathcal{S}IG(\mathcal{C})$ the function $\Phi(f) \in \mathcal{C}$. As usual we assume that there is a constant O for 0 and a unary function symbol S for the successor function. Numerals are defined by $\underline{0} := O$; $\underline{m+1} := S(\underline{m})$. (In case of dyadic strings one has to define formal analogues of $1-2$ strings in the natural way.) For each function symbol $f \in \mathcal{S}IG(\mathcal{C}) \setminus \{0, S\}$ the rewrite system $R_{\mathcal{C}}$ contains a rule

$$f(\vec{x}) \rightarrow_{R_{\mathcal{C}}} t(\lambda\vec{y}.f(\vec{y}), \vec{x}).$$

In all non pathological cases the rewrite system $R_{\mathcal{C}}$ will be terminating and confluent (on ground terms). A successful analysis of $R_{\mathcal{C}}$ will give intrinsic information on (the computational complexity of the functions in) $\mathcal{C}$. For illustration, put for each l-ary function symbol $f \in \mathcal{S}IG(\mathcal{C})$
$\mathcal{D}_{R_{\mathcal{C}},f}(m_1, \ldots, m_l) :\simeq$
$\max\{n : \exists t_1, \ldots, t_n \in \mathcal{G}(\mathcal{S}IG(\mathcal{C})) : t_1 \rightarrow_{R_{\mathcal{C}}} \cdots \rightarrow_{R_{\mathcal{C}}} t_n \ \&\ t_1 = f(\underline{m_1}, \ldots, \underline{m_l})\}$
where $\mathcal{G}(\mathcal{S}IG(\mathcal{C}))$ denotes the set of ground terms over $\mathcal{S}IG(\mathcal{C})$.
Then $\mathcal{D}_{R_{\mathcal{C}},f}(m_1, \ldots, m_l)$, if it is defined, measures the maximal possible lengths of a rewrite computation ending in the numeral $\underline{\Phi(f)(m_1, \ldots, m_l)}$.

Assume that $\mathcal{C}'$ is another complexity class which is for the moment assumed to be closed under "elementary recursive in" and that we want to prove for example that $\mathcal{C} \subseteq \mathcal{C}'$. Then it suffices to show that $\mathcal{D}_{R_{\mathcal{C}},f}$ is bounded in $\mathcal{C}'$, since then each function $\Phi(f)$, which is computed by $R_{\mathcal{C}}$, is computable in $\mathcal{C}' - TIME$ and hence is an element of $\mathcal{C}'$. In Cichon and Weiermann 1995 [6] it is shown how this method can be used in a straightforward way for showing that the primitive recursive functions are closed under parameter recursion, simple nested recursion, unnested multiple recursion and related schemata.

Even in the case that we know $\mathcal{C} \subseteq \mathcal{C}'$ in advance, the approach proposed above yields the interesting information, that *any* non pathological rewrite based algorithm, for example a head reduction or a inside first reduction algorithm, for a function in $\mathcal{C}$ will run in $\mathcal{C}' - TIME$ and yields an element of $\mathcal{C}'$.

In the meantime this approach has been applied successfully to large complexity classes like the elementary recursive functions, the primitive recursive functions, the multiply recursive functions, the ordinal recursive functions, functions which are defined via nested or unnested $\prec$-recursion, and the functions definable in Gödes's T [cf. for example [6, 8, 11, 14, 15, 16]].

It seems quite natural to ask if the term rewriting approach can also be applied to the so called small complexity classes and a contribution to this problem setting will be the goal of this article. We will provide some precise classifications of the computational complexities of algorithms which are based on natural rewriting strategies. It turns out that – in contrast to the large complexity classes – in general we *cannot* expect that $\mathcal{D}_{R_\mathcal{C},f}$ will be bounded in a feasible way even if $\mathcal{C}$ is a small complexity class like the polytime functions. This phenomenon reflects the practical experience that a bad choice of the evaluation strategy in an implementation of a rewrite algorithm may yield an infeasible running time. Nevertheless, on the positive side it turns out that in some interesting cases – for example in case of the rewriting framework for the Bellantoni Cook schemata of predicative recursion – it is possible to introduce a natural restriction ${R_\mathcal{C}}'$ of $R_\mathcal{C}$ so that the resulting derivation lengths $\mathcal{D}_{{R_\mathcal{C}}',f}$ are bounded in a feasible way. Thus even the slowest possible non pathological ${R_\mathcal{C}}'$-reduction strategy computes any $\Phi(f) \in \mathcal{C}$ via ${R_\mathcal{C}}'$ in a feasible time. Therefore, and this seems to be of interest for practical implementations, there is a great flexibility in programming algorithms for the functions in $\mathcal{C}$. In particular it is possible to write a feasible program without having knowledge of the fastest algorithm.

The presentation of this paper is intended to cover a wide range of small complexity classes. For expository reasons we concentrate on the complexity class $\mathcal{C} = \mathcal{P}OLYTIME$, the class of polynomial time computable functions. [The methods developed in this paper, however, apply also to other complexity classes, for example, the class $\mathcal{L}INSPACE$ of functions computable in linear space. We conjecture that the methods developed in this paper can also be applied without much extra complication for derivation lengths classifications of various interesting typed and untyped systems for predicative recursion which have been comprehensively investigated by Leivant and others (cf., for example, [10]). A derivation lengths classification for primitive recursive functionals of finite type has already been obtained in [16].] It turns out that the Bellantoni Cook schemata of predicative recursion are very convenient for the rewrite analysis since they do not refer to principles which depend on bounding functions.

In section 2 we give a formal definition of the canonical rewrite system R_B for these schemata and we prove termination and confluence (on ground terms) for this rewrite system. We also introduce a restricted system ${R_B}'$ which computes the same functions as R_B. In ${R_B}'$ rewriting rules only apply to terms in which all safe arguments are numerals. In section 3 we prove tight exponential bounds (in a polynomial in the lengths of the inputs) on the R_B-derivation lengths. It is shown that the canonical head reduction strategy yields an algorithm which runs in a time which is exponential (in a polynomial in the lengths of the inputs). In section 4 we prove tight polynomial bounds (in the lengths of the inputs) for the ${R_B}'$-derivation lengths and we prove that any canonical head reduction strategy as well as any inside first reduction strategy yield algorithms which run in polynomial time. In section 5 we reprove a non trivial closure property of

$\mathcal{POLYTIME}$ - which is due to Bellantoni - using the term rewriting approach.

The analyses of section 3 and 4 shed some additional light on the different rôles of normal and safe arguments in the Bellantoni Cook schemata of predicative recursion. (Further interesting applications of using the concept of normal and safe arguments can be found in Simmons' article [13].) Following Bellantoni's exposition, the safe arguments ran over "impredicative values" which are known to exist because we "assume the existence of an impredicative defined set $\mathbb{N}$". In the usual machine models this impredicativity of safe arguments is controlled by "performing only operations on them which are constant-time with respect to their size." In the present approach it turns out that it is a practical necessity for implementing an algorithm, which runs in feasible time, to evaluate these impredicative arguments as soon as possible. Therefore calls for safe arguments should be implemented as *call by value* calls. Calls for normal arguments can be implemented as *call by name* calls without disturbing the feasibility of the running time of the resulting algorithm. Thus, the difference of normal and safe arguments is also mirrored in the flexibility of feasible evaluation possibilities. Our exposition is self-contained. No special experience with term rewriting systems is assumed. An introduction into rewrite systems can be found, for example, in [7]. We assume that the reader is familiar with the Bellantoni Cook characterization [3, 4] of $\mathcal{POLYTIME}$.

2 A rewrite system for predicative recursion

Let $\mathbb{N} = \{0, 1, 2, \ldots\}$ be the set of natural numbers. For $i \in \{1, 2\}$ we denote the dyadic successor function $m \mapsto 2{\cdot}m{+}i$ by $\mathcal{S}_i$. Every natural number can be built up from 0 by iterated applications of $\mathcal{S}_1$ and $\mathcal{S}_2$, since every non zero natural number m can be written uniquely as $\sum_{l=1}^{n} 2^{l-1} \cdot d_l$, where $d_l \in \{1, 2\}$ for $1 \le l \le n$. The *dyadic length* of a natural number m, $|m|$ is defined recursively by $|0| := 0$; $|\mathcal{S}_i(m)| := |m| + 1$.

Let k, l be natural numbers. For any $\langle m_1, \ldots, m_k \rangle \in \mathbb{N}^k$ and any $\langle n_1, \ldots, n_l \rangle \in \mathbb{N}^l$ let $\langle m_1, \ldots, m_k; n_1, \ldots, n_l \rangle := \langle \langle m_1, \ldots, m_k \rangle, \langle n_1, \ldots, n_l \rangle \rangle$. We write $\mathbb{N}^{k,l}$ for $\mathbb{N}^k \times \mathbb{N}^l$. For $\mathcal{F} : \mathbb{N}^{k,l} \to \mathbb{N}$ we write $\mathcal{F}(m_1, \ldots, m_k; n_1, \ldots, n_l)$ instead of $\mathcal{F}(\langle \langle m_1, \ldots, m_k \rangle; \langle n_1, \ldots, n_l \rangle \rangle)$.

For any $i \in \{1, 2\}$ we denote by $\mathcal{S}_i^{0,1}$ the number-theoretic function $\langle ; m \rangle \mapsto \mathcal{S}_i(m)$. We denote by $\mathcal{O}^{k,l}$ the number-theoretic function $\langle m_1, \ldots, m_k; n_1, \ldots, n_l \rangle \mapsto 0$. For any k, l, r so that $1 \le r \le k + l$ we denote by $\mathcal{U}_r^{k,l}$ the number-theoretic function $\langle m_1, \ldots, m_k; m_{k+1}, \ldots, m_{k+l} \rangle \mapsto m_r$. By $\mathcal{P}^{0,1}$ we denote the unique number-theoretic function $\mathcal{F} : \mathbb{N}^{0,1} \to \mathbb{N}$ which satisfies $\mathcal{F}(;0) := 0$; $\mathcal{F}(;2 \cdot m + 1) = m$; $\mathcal{F}(;2 \cdot m + 2) = m$. By $\mathcal{C}^{0,3}$ we denote the unique function $\mathcal{F} : \mathbb{N}^{0,3} \mapsto \mathbb{N}$ which satisfies $\mathcal{F}(;0, m_1, m_2) := m_2$; $\mathcal{F}(;2 \cdot m + 1, m_1, m_2) = m_1$; $\mathcal{F}(;2 \cdot m + 2, m_1, m_2) = m_2$. If $k, k', l, l' \in \mathbb{N}$ and $\mathcal{F} : \mathbb{N}^{k',l'} \to \mathbb{N}$ and $\mathcal{G}_1 : \mathbb{N}^{k,0} \to \mathbb{N}, \ldots, \mathcal{G}_{k'} : \mathbb{N}^{k,0} \to \mathbb{N}$ and $\mathcal{H}_1 : \mathbb{N}^{k,l} \to \mathbb{N}, \ldots, \mathcal{H}_{l'} : \mathbb{N}^{k,l} \to \mathbb{N}$ then $\mathcal{SUB}_{k',l'}^{k,l}[\mathcal{F}, \mathcal{G}_1, \ldots, \mathcal{G}_{k'}, \mathcal{H}_1, \ldots, \mathcal{H}_{l'}]$ denotes the

number-theoretic function $\langle m_1,\ldots,m_k;n_1,\ldots,n_l\rangle \mapsto \mathcal{F}(\mathcal{G}_1(m_1,\ldots,m_k;),\ldots,$ $\mathcal{G}_{k'}(m_1,\ldots,m_k;);\mathcal{H}_1(m_1,\ldots,m_k;n_1,\ldots,n_l),\ldots,\mathcal{H}_{l'}(m_1,\ldots,m_k;n_1,\ldots,n_l))$.
If $k,l\in\mathbb{N}$ and $\mathcal{G}:\mathbb{N}^{k,l}\to\mathbb{N}$ and $\mathcal{H}_1:\mathbb{N}^{k+1,l+1}\to\mathbb{N}$ and $\mathcal{H}_2:\mathbb{N}^{k+1,l+1}\to\mathbb{N}$ then $\mathcal{PREC}^{k+1,l}[\mathcal{G},\mathcal{H}_1,\mathcal{H}_2]$ denotes the unique number-theoretic function $\mathcal{F}:\mathbb{N}^{k+1}\times\mathbb{N}^l\to\mathbb{N}$ so that
$\mathcal{F}(0,m_1,\ldots,m_k;n_1,\ldots,n_l)=\mathcal{G}(m_1,\ldots,m_k;n_1,\ldots,n_l)$,
$\mathcal{F}(2\cdot m+1;m_1,\ldots,m_k:n_1,\ldots,n_l)=\mathcal{H}_1(m,m_1,\ldots,m_k;n_1,\ldots,n_l,$
$\mathcal{F}(m,m_1,\ldots,m_k;n_1,\ldots,n_l))$,
$\mathcal{F}(2\cdot m+2;m_1,\ldots,m_k:n_1,\ldots,n_l)=\mathcal{H}_2(m,m_1,\ldots,m_k;n_1,\ldots,n_l,$
$\mathcal{F}(m,m_1,\ldots,m_k;n_1,\ldots,n_l))$.

Definition 2.1 *For any $k,l\in\mathbb{N}$ we define a set $\mathcal{B}\subseteq\{\mathcal{F}:\mathbb{N}^{k,l}\to\mathbb{N}\}$ of number-theoretic functions inductively as follows.*

1. $\mathcal{S}_i^{0,1}\in\mathcal{B}^{0,1}$ *for any* $i\in\{1,2\}$.
2. $\mathcal{O}^{k,l}\in\mathcal{B}^{k,l}$ *for all* $k,l\in\mathbb{N}$.
3. $\mathcal{U}_r^{k,l}\in\mathcal{B}^{k,l}$ *for all* $k,l,r\in\mathbb{N}$ *so that* $1\le r\le k+l$.
4. $\mathcal{P}^{0,1}\in\mathcal{B}^{0,1}$.
5. $\mathcal{C}^{0,3}\in\mathcal{B}^{0,3}$.
6. *If* $k,k',l,l'\in\mathbb{N}$, $\mathcal{F}\in\mathcal{B}^{k',l'}$, $\mathcal{G}_1\in\mathcal{B}^{k,0},\ldots,\mathcal{G}_{k'}\in\mathcal{B}^{k,0}$, *and* $\mathcal{H}_1\in\mathcal{B}^{k,l},\ldots,\mathcal{H}_{l'}\in\mathcal{B}^{k,l}$ *then* $\mathcal{SUB}_{k',l'}^{k,l}[\mathcal{F},\mathcal{G}_1,\ldots,\mathcal{G}_{k'},\mathcal{H}_1,\ldots,\mathcal{H}_{l'}]\in\mathcal{B}^{k,l}$.
7. *If* $k,l\in\mathbb{N}$ *and* $\mathcal{G}\in\mathcal{B}^{k,l}$, $\mathcal{H}_1\in\mathcal{B}^{k+1,l+1}$ *and* $\mathcal{H}_2\in\mathcal{B}^{k+1,l+1}$ *then* $\mathcal{PREC}^{k+1,l}[\mathcal{G},\mathcal{H}_1,\mathcal{H}_2]\in\mathcal{B}^{k+1,l}$

$\mathcal{B}:=\bigcup_{k,l\in\mathbb{N}}\mathcal{B}^{k,l}$ *is the set of* predicative recursive functions.

Bellantoni has shown in [4] that $\mathcal{B}$ coincides with the complexity class $\mathcal{POLYTIME}$ of polynomial time computable functions.

Definition 2.2 *For any $k,l\in\mathbb{N}$ we define a set $B^{k,l}$ of function symbols inductively as follows.*

1. $S_i^{0,1}\in B^{0,1}$ *for* $i\in\{0,1\}$.
2. $O^{k,l}\in B^{k,l}$.
3. $U_r^{k,l}\in B^{k,l}$ *for any* $1\le r\le k+l$.
4. $P^{0,1}\in B^{0,1}$.
5. $C^{0,3}\in B^{0,3}$.
6. *If* $k',l'\in\mathbb{N}$ *and* $f\in B^{k',l'}$ *and* $g_1,\ldots,g_{k'}\in B^{k,0}$ *and* $h_1,\ldots,h_{l'}\in B^{k,l}$ *then* $SUB_{k',l'}^{k,l}[f,g_1,\ldots,g_{k'},h_1,\ldots,h_{l'}]\in B^{k,l}$.

7. *If* $g \in B^{k,l}$ *and* $h_1, h_2 \in B^{k+1,l+1}$ *then* $PREC^{k+1,l}[g, h_1, h_2] \in B^{k+1,l}$.

Let $B := \bigcup_{k,l \in \mathbb{N}} B^{k,l}$ *be the set of* predicative recursive function symbols.

We often write O for $O^{0,0}$ and S_i for $S_i^{0,1}$ for $i \in \{1, 2\}$.

Definition 2.3 *For any* $f \in B$ *we define recursively the length of* f*,* $lh(f)$*, as follows.*

1. $lh(S_i^{0,1}) := 1$ *for* $i \in \{1, 2\}$.
2. $lh(O^{k,l}) := 1$.
3. $lh(U_r^{k,l}) := 1$ *for any* $1 \leq r \leq k + l$.
4. $lh(P^{0,1}) := 1$.
5. $lh(C^{0,3}) := 1$.
6. $lh(SUB_{k',l'}^{k,l}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]) := 1 + lh(f) + lh(g_1) + \ldots + lh(g_{k'}) + lh(h_1) + \ldots + lh(h_{l'})$.
7. $lh(PREC^{k+1,l}[g, h_1, h_2]) := 1 + lh(g) + lh(h_1) + lh(h_2)$.

Definition 2.4 *For any* $f \in B$ *we define recursively* $\Phi(f) \in \mathcal{B}$ *as follows.*

1. $\Phi(S_i^{0,1}) := \mathcal{S}_i^{0,1}$ *for* $i \in \{1, 2\}$.
2. $\Phi(O^{k,l}) := \mathcal{O}^{k,l}$.
3. $\Phi(U_r^{k,l}) := \mathcal{U}_r^{k,l}$ *for any* $1 \leq r \leq k + l$.
4. $\Phi(P^{0,1}) := \mathcal{P}^{0,1}$.
5. $\Phi(C^{0,3}) := \mathcal{C}^{0,3}$.
6. $\Phi(SUB_{k',l'}^{k,l}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]) :=$
   $\mathcal{S}UB_{k',l'}^{k,l}[\Phi(f], \Phi(g_1), \ldots, \Phi(g_{k'}), \Phi(h_1), \ldots, \Phi(h_{l'}))$.
7. $\Phi(PREC^{k+1,l}[g, h_1, h_2]) := \mathcal{P}REC^{k+1,l}[\Phi(g], \Phi(h_1), \Phi(h_2))$.

Lemma 2.5 *For any* $\mathcal{F} \in \mathcal{B}^{k,l}$ *there exists an* $f \in B^{k,l}$ *so that* $\mathcal{F} = \Phi(f)$.

Proof. The assertion is easy to see. □

Definition 2.6 *Let* X *be a countable infinite set of variables which is disjoint from* B*. We define the set of predicative recursive terms,* $T(B)$*, over the signature* B *inductively as follows.*

1. *If $x \in X$ then $x \in T(B)$.*

2. *If $k, l \in \mathbb{N}$ and $f \in B^{k,l}$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ then $f(s_1, \ldots, s_k; t_1, \ldots, t_l) \in T(B)$.*

We denote the set of variables, which occur in $t \in T(B)$, by $FV(t)$. A term t is called *ground*, if $FV(t) = \emptyset$. The set of ground terms is denoted by $G(B)$.
For any $t \in G(B)$ we define $\Phi(t) \in \mathbb{N}$ recursively as follows. If $k, l \in \mathbb{N}$, $f \in B^{k,l}$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in G(B)$ then $\Phi(f(s_1, \ldots, s_k; t_1, \ldots, t_l)) := \Phi(f)(\Phi(s_1), \ldots, \Phi(s_k); \Phi(t_1), \ldots, \Phi(t_l))$.

For every natural number m we define recursively a *numeral* $\underline{m}$ as follows:

1. $\underline{0} := O$.

2. $\underline{\mathcal{S}_1^{0,1}(;m)} := S_1(;\underline{m})$.

3. $\underline{\mathcal{S}_2^{0,1}(;m)} := S_2(;\underline{m})$.

Then $\underline{m} \in T(B)$ and $\Phi(\underline{m}) = m$ for any $m \in \mathbb{N}$. For $t \in T(B)$ we define the length of t, $lh(t)$, recursively as follows. If $k, l \in \mathbb{N}$, $f \in B^{k,l}$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ then $lh(f(s_1, \ldots, s_k; t_1, \ldots, t_l)) := lh(f) + lh(s_1) + \ldots + lh(s_k) + lh(t_1) + \ldots + lh(t_l)$. A mapping $\sigma : X \to T(B)$ is called *substitution*. For any $t \in T(B)$ we define $t\sigma$ recursively as follows:

1. $x\sigma := \sigma(x)$.

2. $t\sigma := f(s_1\sigma, \ldots, s_k\sigma; t_1\sigma, \ldots, t_l\sigma)$ if $t = f(s_1, \ldots, s_k; t_1, \ldots, t_l)$.

Then $t\sigma \in T(B)$ for any $t \in T(B)$.

A *rewrite system* R over B is a set consisting of ordered pairs of elements from $T(B)$. The rewrite relation $\to_R$ is the least binary relation on $T(B)$ so that

1. If $\langle l, r \rangle \in R$ and $\sigma : X \to T(B)$ then $l\sigma \to_R r\sigma$.

2. If $k, l, r \in \mathbb{N}$ and $1 \le r \le k$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ and if $s_r \to_R s'_r$ then $f(s_1, \ldots, s_r, \ldots, s_k; t_1, \ldots, t_l) \to_R f(s_1, \ldots, s'_r, \ldots, s_k; t_1, \ldots, t_l)$

3. If $k, l, r \in \mathbb{N}$ and $1 \le r \le l$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ and if $t_r \to_R t'_r$ then $f(s_1, \ldots, s_k; t_1, \ldots, t_r, \ldots, t_l) \to_R f(s_1, \ldots, s_k; t_1, \ldots, t'_r, \ldots, t_l)$.

R is called *terminating* if there is no infinite sequence $(t_n)_{n \in \mathbb{N}}$ of elements in $T(B)$ so that $t_n \to_R t_{n+1}$ holds for all $n \in \mathbb{N}$. Let $\to_R^*$ be the transitive reflexive closure of $\to_R$. R is called *confluent* if for all $s, t, u \in T(B)$ so that $s \to_R^* t$ and $s \to_R^* u$ there is a $v \in T(B)$ so that $t \to_R^* v$ and $u \to_R^* v$. R is called *confluent on ground terms* if for all $s, t, u \in G(B)$ so that $s \to_R^* t$ and $s \to_R^* u$ there is a $v \in G(B)$ so that $t \to_R^* v$ and $u \to_R^* v$. A term $t \in T(B)$ is called a *normal form* if there is no $u \in T(B)$ so that $t \to_R u$. It is easy to see that, if R is terminating and confluent on ground terms then for every $t \in G(B)$ there is a unique normal form $u \in G(B)$ so that $t \to_R^* u$.

For any $s, t \in T(B)$ we denote by $s \to t$ the ordered pair $\langle s, t \rangle$.

Definition 2.7 *Definition of a set R_B of non feasible rewrite rules for predicative recursion.*

1. $O^{k,l}(x_1, \ldots, x_k; y_1, \ldots, y_l) \to O$ *for any* $k, l \in \mathbb{N}$ *so that* $k + l > 0$.
2. $U_r^{k,l}(x_1, \ldots, x_k; x_{k+1}, \ldots, x_{k+l}) \to x_r$ *for any* $1 \le r \le k + l$.
3. $P^{0,1}(; O) \to O$.
4. $P^{0,1}(; S_i(; y)) \to y$ *for* $i \in \{1, 2\}$.
5. $C^{0,3}(; O, y_1, y_2) \to y_1$.
6. $C^{0,3}(; S_i(; y), y_1, y_2) \to y_i$ *for* $i \in \{1, 2\}$.
7. $SUB^{k,l}_{k',l'}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}](x_1, \ldots, x_k; y_1, \ldots, y_l) \to$
   $f(g_1(x_1, \ldots, x_k;), \ldots, g_{k'}(x_1, \ldots, x_k;);$
   $h_1(x_1, \ldots, x_k; y_1, \ldots, y_l), \ldots, h_{l'}(x_1, \ldots, x_k; y_1, \ldots, y_l))$.
8. $PREC^{k+1,l}[g, h_1, h_2](O, x_1, \ldots, x_k; y_1, \ldots, y_l) \to$
   $g(x_1, \ldots, x_k; y_1, \ldots, y_l)$.
9. $PREC^{k+1,l}[g, h_1, h_2](S_i(; x), x_1, \ldots, x_k; y_1, \ldots, y_l) \to$
   $h_i(x, x_1, \ldots, x_k; y_1, \ldots, y_l, PREC^{k+1,l}[g, h_1, h_2](x, x_1, \ldots, x_k; y_1, \ldots, y_l))$
   for any $i \in \{1, 2\}$.

Lemma 2.8 *R_B is terminating.*

Proof. The rules of R_B are reducing under the multiset path ordering. To see this we define as a suitable well-founded precedence $\succ$ on B

$$f \succ g \quad :\Longleftrightarrow \quad lh(f) > lh(g).$$

Then the induced multiset path ordering $\succ_{mpo}$ on $T(B)$ is well-founded [7, 9]. It is easily shown that $l \succ_{mpo} r$ holds for all rules $l \to r$ of R_B.

An alternative termination proof is given in section 3. □

Lemma 2.9 *R_B is confluent on ground terms.*

Proof. R_B is orthogonal, i.e. R_B does not contain critical pairs. Termination of R_B then yields confluence (in general).

Alternatively, we could argue directly as follows. R_B is terminating. The rules of R_B respect the intended semantics given by the evaluation function Φ. Every ground normal form is a numeral. Putting things together the assertion follows. □

Definition 2.10 *Definition of a set $R_B{}'$ of feasible rewrite rules for predicative recursion.*

1. $O^{k,l}(x_1,\ldots,x_k;y_1,\ldots,y_l) \to O$ *for any* $k,l \in \mathbb{N}$ *so that* $k+l>0$.
2. $U_r^{k,l}(x_1,\ldots,x_k;x_{k+1},\ldots,x_{k+l}) \to x_r$ *for any* $1 \le r \le k+l$.
3. $P^{0,1}(;O) \to O$.
4. $P^{0,1}(;S_i(;y)) \to y$ *for* $i \in \{1,2\}$.
5. $C^{0,3}(;O,y_1,y_2) \to y_1$.
6. $C^{0,3}(;S_i(;y),y_1,y_2) \to y_i$ *for* $i \in \{1,2\}$.
7. $SUB^{k,l}_{k',l'}[f,g_1,\ldots,g_{k'},h_1,\ldots,h_{l'}](x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}) \to$ $f(g_1(x_1,\ldots,x_k;),\ldots,g_{k'}(x_1,\ldots,x_k;);$ $h_1(x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}),\ldots,h_{l'}(x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}))$ *for any sequence* $\langle\underline{n_1},\ldots,\underline{n_l}\rangle$ *of numerals.*
8. $PREC^{k+1,l}[g,h_1,h_2](O,x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}) \to$ $g(x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l})$ *for any sequence* $\langle\underline{n_1},\ldots,\underline{n_l}\rangle$ *of numerals.*
9. $PREC^{k+1,l}[g,h_1,h_2](S_i(;x),x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}) \to$ $h_i(x,x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l},PREC^{k+1,l}[g,h_1,h_2](x,x_1,\ldots,x_k;\underline{n_1},\ldots,\underline{n_l}))$ *for any* $i \in \{1,2\}$ *and any sequence* $\langle\underline{n_1},\ldots,\underline{n_l}\rangle$ *of numerals.*

Lemma 2.11 *Let* $s,t \in T(B)$. *If* $s \to_{R'_B} t$ *then* $s \to_{R_B} t$.

Proof. This follows from the definition. □

Lemma 2.12 $R_B{}'$ *is terminating.*

Proof. This follows from the termination of R_B and Lemma 2.11. □

Lemma 2.13 $R_B{}'$ *is confluent on ground terms.*

Proof. $R_B{}'$ is orthogonal. Termination of $R_B{}'$ then yields confluence (in general).

Alternatively, we could argue directly as follows. $R_B{}'$ is terminating. The rules of $R_B{}'$ respect the intended semantics given by the evaluation function Φ. Every ground normal form is a numeral. Putting things together the assertion follows. □

3 A number-theoretic interpretation for the non-feasible rewrite schemes of predicative recursion

In this section it is shown that the R_B-derivation lengths are bounded exponentially (in a polynomial in the lengths of the inputs). It turns out that in the worst case the freedom to evaluate subterms in any order leads to a derivation tree (i.e. the tree obtained by parallelizing reductions whenever possible) that has exponential size.

Definition 3.1 *For $f \in B^{k,l}$ we define the derivation length function $\mathcal{D}_{R_B,f}$ as follows: $\mathcal{D}_{R_B,f}(m_1,\dots,m_k;n_1,\dots,n_l) := max\{n|\exists t_1,\dots,t_n \in G(B) : t_1 \rightarrow_{R_B} \dots \rightarrow_{R_B} t_n \ \& \ t_1 = f(\underline{m_1},\dots,\underline{m_k};\underline{n_1},\dots,\underline{n_l})\}$.*

Since R_B is terminating under the multiset path ordering, we know in advance by Hofbauer's result [9] that for any $f \in B^{k,l}$ the derivation length function $\mathcal{D}_{R_B,f}$ is bounded by a primitive recursive function. In fact, it can be shown by a simple calculation that for any $f \in B^{k,l}$ the derivation length function $\mathcal{D}_{R_B,f}$ is bounded by an elementary recursive function. In this section we prove tight bounds on $\mathcal{D}_{R_B,f}$.

Let us fix monotone polynomials q_f, i.e. polynomials with nonnegative coefficients, for every $f \in B$:

1. For $f = O^{k,l}, U_r^{k,l}$ with $1 \le r \le k+l$, $S_i^{0,1}, P^{0,1}, C^{0,3}$, where $f \in B^{k,l}$, we set $q_f(m_1,\dots,m_k) = 1 + \sum_{d=1}^{k} m_d$.
2. For $f = SUB^{k,l}_{k',l'}[\tilde{f}, g_1,\dots,g_{k'},h_1,\dots,h_{l'}]$ we set $q_f(m_1,\dots,m_k) = q_{\tilde{f}}\Big(q_{g_1}(m_1,\dots,m_k),\dots,q_{g_{k'}}(m_1,\dots,m_k)\Big) + \sum_{d=1}^{l'} q_{h_d}(m_1,\dots,m_k)$.
3. For $f = PREC^{k+1,l}[g,h_1,h_2]$ we set $q_f(m,m_1,\dots,m_k) = m \cdot \Big(q_{h_1}(m,m_1,\dots,m_k) + q_{h_2}(m,m_1,\dots,m_k)\Big) + q_g(m_1,\dots,m_k)$

As in [4] the following polymax bounding lemma is proved:

Lemma 3.2

$$|\Phi(f)(m_1,\dots,m_k;n_1,\dots,n_l)| \le q_f(|m_1|,\dots,|m_k|) + \max_{1\le d\le l} |n_d|$$

for $f \in B^{k,l}$ and $m_1,\dots,m_k,n_1,\dots,n_l \in \mathbb{N}$. □

Let Ψ be an unary function and $u_1,\dots,u_k$ be a sequence of elements in the domain of Ψ. In the sequel we write $\Psi(\vec{u})$ for $\Psi(u_1),\dots,\Psi(u_k)$. For example $|\Phi(\vec{u})|$ abbreviates $|\Phi(u_1)|,\dots,|\Phi(u_k)|$.

For $f \in B^{k,l}$ we will define a number-theoretic interpretation $\mathrm{I}_0(f) : \mathbb{N}^k \longrightarrow \mathbb{N}$ which is a monotone polynomial. Then the interpretation of the terms in $T(B)$ is defined as follows. For a variable $x \in X$ we set $\mathrm{I}(x) = 1$. For $f \in B^{k,l}$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ we define

$$\mathrm{I}(f(s_1, \ldots, s_k; t_1, \ldots, t_l)) := 2^{\mathrm{I}_0(f)\left(|\Phi(\vec{s})|\right)} \cdot \Big(\sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(f)\Big).$$

Obviously we have $\mathrm{I}(f(\vec{s}, \vec{t})) \geq \sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(f) \geq 1$. By induction on $lh(t)$ we get $\mathrm{I}(t) \geq lh(t)$ for any $t \in T(B)$.

Definition 3.3 *Recursive definition of* $\mathrm{I}_0(f)$ *for* $f \in B$.

1. *For* $f = O^{k,l}, U_r^{k,l}$ *with* $1 \leq r \leq k + l$, $S_i^{0,1}, P^{0,1}, C^{0,3}$, *where* $f \in B^{k,l}$, *we set* $\mathrm{I}_0(f)(m_1, \ldots, m_k) = 0$

2. $\mathrm{I}_0\Big(SUB_{k',l'}^{k,l}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]\Big)(m_1, \ldots, m_k)$

$$= \mathrm{I}_0(f)\Big(q_{g_1}(\vec{m}), \ldots, q_{g_{k'}}(\vec{m})\Big) + \sum_{d=1}^{k'} \mathrm{I}_0(g_d)(\vec{m}) + \sum_{d=1}^{l'} \mathrm{I}_0(h_d)(\vec{m}) + 1$$

3. $\mathrm{I}_0\Big(PREC^{k+1,l}[g, h_1, h_2]\Big)(m, m_1, \ldots, m_k)$

$$= m \cdot \Big(\mathrm{I}_0(h_1)(m, \vec{m}) + \mathrm{I}_0(h_2)(m, \vec{m})\Big) + \mathrm{I}_0(g)(\vec{m})$$

From the definition we easily see that $\mathrm{I}_0(f)$ is a monotone polynomial for any $f \in B$.

We compute $\mathrm{I}(S_i(;s)) = 2^{\mathrm{I}_0(S_i)} \cdot (\mathrm{I}(s) + lh(S_i)) = \mathrm{I}(s) + 1$, so $\mathrm{I}(\underline{m}) = |m| + 1$ for every numeral $\underline{m}$.

Using $2^m + 2^n \leq 2^{m+n}$ for $m, n \geq 1$ we see

$$\begin{aligned}
&\mathrm{I}\Big(f(\underline{m_1}, \ldots, \underline{m_k}; \underline{n_1}, \ldots, \underline{n_l})\Big) \\
= \;& 2^{\mathrm{I}_0(f)(|\Phi(\underline{\vec{m}})|)} \cdot \Big(\sum_{d=1}^{k} \mathrm{I}(\underline{m}_d) + \sum_{d=1}^{l} \mathrm{I}(\underline{n}_d) + lh(f)\Big) \\
\leq \;& 2^{\mathrm{I}_0(f)(|\vec{m}|)} \cdot \Big(\sum_{d=1}^{k} (|m_d| + 1) + \sum_{d=1}^{l} (|n_d| + 1) + lh(f)\Big) \\
\leq \;& 2^{\mathrm{I}_0(f)(|\vec{m}|) + \sum_{d=1}^{k} |m_d| + \sum_{d=1}^{l} |n_d| + lh(f)}
\end{aligned}$$

Therefore $\mathrm{I}(f(\underline{m_1}, \ldots, \underline{m_k}; \underline{n_1}, \ldots, \underline{n_l}))$ is bounded by an exponential of a monotone polynomial in the lengths of the inputs $m_1, \ldots, m_k, n_1, \ldots, n_l$.

Theorem 3.4 *R_B is terminating and for every $f \in B$ the derivation length function $\mathcal{D}_{R_B,f}$ is bounded by an exponential of a monotone polynomial in the lengths of the inputs.*

This theorem is an immediate consequence of

Lemma 3.5 *If $s, t \in G(B)$ and $s \to_{R_B} t$ then* $\mathrm{I}(s) > \mathrm{I}(t)$.

Proof of Theorem 3.4. If $t_1, \ldots, t_n \in G(B)$ with $t_1 \to_{R_B} \ldots \to_{R_B} t_n$ then Lemma 3.5 implies $n \leq \mathrm{I}(t_1)$. Therefore $\mathcal{D}_{R_B,f}(m_1, \ldots, m_k, n_1, \ldots, n_l) \leq \mathrm{I}(f(\underline{m_1}, \ldots, \underline{m_k}; \underline{n_1}, \ldots, \underline{n_l}))$ and this is bounded by an exponential of a monotone polynomial in the lengths of the inputs. $\Box$

Proof of Lemma 3.5. It is easy to check that $\mathrm{I}(f(\ldots s \ldots)) > \mathrm{I}(f(\ldots t \ldots))$ provided that $\mathrm{I}(s) > \mathrm{I}(t)$. So we only have to prove that $\mathrm{I}(s\sigma) > \mathrm{I}(t\sigma)$ for all $\langle s, t\rangle \in R_B$ and all ground substitutions σ. To do this we only consider the nontrivial cases.

1. Let $\tilde{f} = SUB^{k,l}_{k',l'}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]$. We consider the case

$$\begin{aligned} &\tilde{f}(s_1, \ldots, s_k; t_1, \ldots, t_l) \\ &\quad \to_{R_B} \quad f\big(g_1(\vec{s};), \ldots, g_{k'}(\vec{s};); h_1(\vec{s}; \vec{t}), \ldots, h_{l'}(\vec{s}; \vec{t})\big). \end{aligned}$$

Using $lh(f), lh(g_1), \ldots, lh(g_{k'}), lh(h_1), \ldots, lh(h_{l'}) < lh\big(\tilde{f}\big)$ we compute

$$\begin{aligned} &\mathrm{I}\Big(f\big(g_1(\vec{s};), \ldots, g_{k'}(\vec{s};); h_1(\vec{s};\vec{t}), \ldots, h_{l'}(\vec{s};\vec{t})\big)\Big) \\ &= \quad 2^{\mathrm{I}_0(f)\big(|\Phi(g_1(\vec{s};))|, \ldots, |\Phi(g_{k'}(\vec{s};))|\big)} \cdot \Big(\sum_{d=1}^{k'} 2^{\mathrm{I}_0(g_d)(|\Phi(\vec{s})|)} \cdot \Big[\sum_{d=1}^{k} \mathrm{I}(s_d) + lh(g_d)\Big] \\ &\qquad + \sum_{d=1}^{l'} 2^{\mathrm{I}_0(h_d)(|\Phi(\vec{s})|)} \cdot \Big[\sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(h_d)\Big] + lh(f)\Big) \\ &< \quad 2^{\mathrm{I}_0(f)\big(q_{g_1}(|\Phi(\vec{s})|), \ldots, q_{g_{k'}}(|\Phi(\vec{s})|)\big)} \cdot \Big(\sum_{d=1}^{k'} 2^{\mathrm{I}_0(g_d)(|\Phi(\vec{s})|)} + \sum_{d=1}^{l'} 2^{\mathrm{I}_0(h_d)(|\Phi(\vec{s})|)} + 1\Big) \\ &\qquad \cdot \Big(\sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh\big(\tilde{f}\big)\Big) \\ &\leq \quad 2^{\mathrm{I}_0(f)\big(q_{g_1}(|\Phi(\vec{s})|), \ldots, q_{g_{k'}}(|\Phi(\vec{s})|)\big) + \sum_{d=1}^{k'} \mathrm{I}_0(g_d)(|\Phi(\vec{s})|) + \sum_{d=1}^{l'} \mathrm{I}_0(h_d)(|\Phi(\vec{s})|) + 1} \\ &\qquad \cdot \Big(\sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh\big(\tilde{f}\big)\Big) \\ &= \quad \mathrm{I}\Big(\tilde{f}(\vec{s}; \vec{t})\Big) \end{aligned}$$

2. Let $f = PREC^{k+1,l}[g, h_1, h_2]$ and consider the case

$$\begin{aligned} & f(S_i(;s), s_1, \ldots, s_k; t_1, \ldots, t_l) \\ \to_{R_B} \quad & h_i\Big(s, s_1, \ldots, s_k; t_1, \ldots, t_l, f(s, s_1, \ldots, s_k; t_1, \ldots, t_l)\Big) \end{aligned}$$

for $i \in \{1, 2\}$. Using $lh(h_i) < lh(f)$ we compute:

$$\begin{aligned} & \mathrm{I}\Big(h_i\Big(s, s_1, \ldots, s_k; t_1, \ldots, t_l, f(s, s_1, \ldots, s_k; t_1, \ldots, t_l)\Big)\Big) \\ = \quad & 2^{\mathrm{I}_0(h_i)(|\Phi(s,\vec{s})|)} \cdot \Big(\mathrm{I}(s) + \sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) \\ & + 2^{\mathrm{I}_0(f)(|\Phi(s,\vec{s})|)} \cdot \Big[\mathrm{I}(s) + \sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(f)\Big] + lh(h_i)\Big) \\ < \quad & 2^{\mathrm{I}_0(h_i)(|\Phi(s,\vec{s})|)} \cdot \Big(1 + 2^{\mathrm{I}_0(f)(|\Phi(s,\vec{s})|)}\Big) \\ & \cdot \Big(\mathrm{I}(s) + \sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(f)\Big) \\ \leq \quad & 2^{(\mathrm{I}_0(h_i)(|\Phi(s,\vec{s})|)+1)+|\Phi(s)|\cdot\big(\mathrm{I}_0(h_1)(|\Phi(s,\vec{s})|)+\mathrm{I}_0(h_2)(|\Phi(s,\vec{s})|)\big)+\mathrm{I}_0(g)(|\Phi(\vec{s})|)} \\ & \cdot \Big(\mathrm{I}(s) + \sum_{d=1}^{k} \mathrm{I}(s_d) + \sum_{d=1}^{l} \mathrm{I}(t_d) + lh(f)\Big) \\ \leq \quad & \mathrm{I}\Big(f(S_i(;s), \vec{s}; \vec{t})\Big) \end{aligned}$$

$\square$

In the following we give an example of a function f so that the derivation length function $\mathcal{D}_{R_B,f}$ is not bounded by a polynomial in the lengths of the inputs. To do so we define a function symbol

$$g := PREC^{1,1}[U_1^{0,1}, SUB_{0,2}^{1,2}[U_1^{0,2}, U_3^{1,2}, U_3^{1,2}], SUB_{0,2}^{1,2}[U_1^{0,2}, U_3^{1,2}, U_3^{1,2}]]$$

and terms $T_0(x) := x$ and $T_{n+1}(x) := U_1^{0,2}(; T_n(x), T_n(x))$. The connection between g and the T_n's is

Lemma 3.6 *$g(\underline{m}; s) \to_{R_B}^{*} T_{|m|}(s)$ for all $m \in \mathbb{N}$ and $s \in T(B)$.*

Proof. The assertion is proved by induction on m. First we see $g(\underline{0}; s) \to_{R_B} U_1^{0,1}(; s) \to_{R_B} s = T_{|0|}(s)$. In the induction step we compute

$$\begin{aligned} & g(\underline{\mathcal{S}_i(;m)}; s) = g(S_i(;\underline{m}); s) \\ \to_{R_B} \quad & SUB_{0,2}^{1,2}[U_1^{0,2}, U_3^{1,2}, U_3^{1,2}](\underline{m}; s, g(\underline{m}; s)) \end{aligned}$$

$$
\begin{array}{ll}
\overset{i.h.}{\rightarrow^*_{R_B}} & SUB^{1,2}_{0,2}[U^{0,2}_1, U^{1,2}_3, U^{1,2}_3](\underline{m}; s, T_{|m|}(s)) \\
\rightarrow_{R_B} & U^{0,2}_1\Big(; U^{1,2}_3(\underline{m}; s, T_{|m|}(s)), U^{1,2}_3(\underline{m}; s, T_{|m|}(s))\Big) \\
\rightarrow^2_{R_B} & U^{0,2}_1(; T_{|m|}(s), T_{|m|}(s)) \ = \ T_{|m|+1}(s) \ = \ T_{|\mathcal{S}_i(;m)|}(s)
\end{array}
$$

□

Consider some term s which has some derivation with length greater than 0. Then the following lemma shows that $T_k(s)$ has a derivation with length at least 2^k.

Lemma 3.7 $l_k(m) := \max\{n | \exists t_1, \ldots, t_n \in G(B) : t_1 \rightarrow_{R_B} \ldots \rightarrow_{R_B} t_n \ \& \ t_1 = T_k(O^{1,0}(\underline{m};)) \ \& \ t_n = O\} \geq 2^k$.

Proof. The assertion is proved by induction on k for any m. Let s be the term $O^{1,0}(\underline{m};)$. We see $T_0(s) = s \rightarrow_{R_B} O$, so $l_0(m) \geq 2^0$. For the induction step we compute

$$T_{k+1}(s) = U^{0,2}_1(; T_k(s), T_k(s)) \rightarrow^{l_k(m)}_{R_B} U^{0,2}_1(; 0, T_k(s)) \rightarrow^{l_k(m)}_{R_B} U^{0,2}_1(; O, O) \rightarrow_{R_B} O,$$

so $l_{k+1}(m) > 2 \cdot l_k(m)$ and the induction hypothesis leads to $l_{k+1}(m) \geq 2^{k+1}$. □

We define $f = SUB^{1,0}_{1,1}[g, U^{1,0}_1, O^{1,0}]$ and prove

Theorem 3.8 *$\mathcal{D}_{R_B,f}(m) > 2^{|m|}$ for all $m \in \mathbb{N}$.*

Proof. We give a derivation of $f(\underline{m}) \rightarrow^*_{R_B} O$ which has a length greater than $2^{|m|}$.

$$
\begin{array}{lll}
f(\underline{m}) & \rightarrow_{R_B} & g\Big(U^{1,0}_1(\underline{m};); O^{1,0}(\underline{m};)\Big) \\
& \rightarrow_{R_B} & g\Big(\underline{m}; O^{1,0}(\underline{m};)\Big) \\
& \rightarrow^*_{R_B} & T_{|m|}(O^{1,0}(\underline{m};)) \rightarrow^{l_{|m|}(m)}_{R_B} O.
\end{array}
$$

With Lemma 3.7 this shows $\mathcal{D}_{R_B,f}(m) > l_{|m|}(m) \geq 2^{|m|}$. □

Corollary 3.9 *$\mathcal{D}_{R_B,f}$ is not bounded by a polynomial in the lengths of the inputs.* □

As a matter of fact we know that for every monotone polynomial $p(m)$ there is a polytime function symbol $g_p \in B^{1,0}$ so that $|\Phi(g_p)(m;)| = p(|m|)$ for all $m \in \mathbb{N}$. With Theorem 3.8 we see

$$\mathcal{D}_{R_B, SUB^{1,0}_{1,0}[f,g_p]}(m) \geq \mathcal{D}_{R_B,f}(\Phi(g_p)(m;)) > 2^{|\Phi(g_p)(m;)|} = 2^{p(|m|)}.$$

Corollary 3.10 *Let $k \in \mathbb{N}$ and $f \in B^{k,0}$. Any reasonable R_B–head reduction strategy yields an algorithm for $\Phi(f)$ which runs in exponential time. (More generally, any reasonable R_B reduction strategy yields an algorithm for $\Phi(f)$ which runs in exponential time.)*

Proof. By folklore, there exists a polytime algorithm which, if possible, rewrites any ground term $s \in G(B)$ via an R_B–head reduction step into its reduct. Consider any deterministic head reduction sequence $f(\underline{m_1}, \ldots, \underline{m_k}) = t_1 \rightarrow_{R_B} \cdots \rightarrow_{R_B} t_n = \underline{\Phi(f)(m_1, \ldots, m_k)}$. Then, by Theorem 3.4 and the remark before Definition 3.3, n and the lengths of any t_i for $i \in \{1, \ldots, n\}$ are bounded by an exponential in a polynomial in the dyadic lengths of $m_1, \ldots, m_k$. Putting things together, the assertion follows. □

4 A number-theoretic interpretation for the feasible rewrite schemes of predicative recursion

In this section it is shown that the R'_B-derivation lengths are bounded by a polynomial in the lengths of the inputs. Since in R'_B the reduction system is forced to evaluate safe positions first the size of the derivation tree is polynomial.

Definition 4.1 *For $f \in B^{k,l}$ we define the derivation length function $\mathcal{D}_{R'_B,f}$ as follows: $\mathcal{D}_{R'_B,f}(m_1, \ldots, m_k; n_1, \ldots, n_l) := max\{n | \exists t_1, \ldots, t_n \in G(B) : t_1 \rightarrow_{R'_B} \ldots \rightarrow_{R'_B} t_n \ \& \ t_1 = f(\underline{m_1}, \ldots, \underline{m_k}; \underline{n_1}, \ldots, \underline{n_l})\}$.*

For $f \in B^{k,l}$ we will define a number-theoretic interpretation $\mathrm{J}_0(f) : \mathbb{N}^k \longrightarrow \mathbb{N}$ which is a monotone polynomial. Then the interpretation of the terms in $T(B)$ is defined as follows. For a variable $x \in X$ we set $\mathrm{J}(x) = 1$. For $f \in B^{k,l}$ and $s_1, \ldots, s_k, t_1, \ldots, t_l \in T(B)$ we define

$$\begin{aligned} \mathrm{J}(f(\vec{s};\vec{t})) \quad &:= \quad \mathrm{J}_0(f)\big(|\Phi(\vec{s})|\big) \cdot \Big(\sum_{d=1}^{k} \mathrm{J}(s_d) + \max_{1 \le d \le l} |\Phi(t_d)| + 1 \Big) \\ &\qquad + \sum_{d=1}^{l} \mathrm{J}(t_d) + lh(f). \end{aligned}$$

Definition 4.2 *Recursive definition of $\mathrm{J}_0(f)$ for $f \in B$.*

1. *For $f = O, S_i^{0,1}, P^{0,1}, C^{0,3}$ we set $\mathrm{J}_0(f) = 0$*
2. *For $f = O^{k,l}, U_r^{k,l}$, where $1 \le r \le k+l$, we set $\mathrm{J}_0(f)(m_1, \ldots, m_k) = 1$*
3. $\mathrm{J}_0\Big(SUB^{k,l}_{k',l'}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]\Big)(m_1, \ldots, m_k)$

$$= \quad \mathrm{J}_0(f)\Big(q_{g_1}(\vec{m}), \ldots, q_{g_{k'}}(\vec{m})\Big) \cdot \Big(\sum_{d=1}^{k'} [\mathrm{J}_0(g_d)(\vec{m}) + lh(g_d)]$$

$$+\sum_{d=1}^{l'} q_{h_d}(\vec{m}) + 1\Big)$$

$$+\sum_{d=1}^{l'} \mathrm{J}_0(h_d)(\vec{m}) + l \cdot l'$$

4. $\mathrm{J}_0\Big(PREC^{k+1,l}[g, h_1, h_2]\Big)(m, m_1, \ldots, m_k)$

$$= m \cdot \Big([\mathrm{J}_0(h_1)(m, \vec{m}) + \mathrm{J}_0(h_2)(m, \vec{m})] \cdot q_f(m, \vec{m}) + l + \max\{lh(h_1), lh(h_2)\}\Big) + \mathrm{J}_0(g)(\vec{m})$$

From the definition we easily see that $\mathrm{J}_0(f)$ is a monotone polynomial for any $f \in B$. Obviously we have $\mathrm{J}(f(\vec{s}, \vec{t})) \geq \sum_{d=1}^{k} \mathrm{J}(s_d) + \sum_{d=1}^{l} \mathrm{J}(t_d) + lh(f) \geq 1$. By induction on $lh(t)$ we get $\mathrm{J}(t) \geq lh(t)$ for any $t \in T(B)$.

We compute $\mathrm{J}(S_i(;s)) = \mathrm{J}_0(S_i) \cdot (|\Phi(s)| + 1) + \mathrm{J}(s) + 1 = \mathrm{J}(s) + 1$, so $\mathrm{J}(\underline{m}) = |m| + 1$ for every numeral $\underline{m}$.

We see that $\mathrm{J}(f(\underline{m_1}, \ldots, \underline{m_k}, \underline{n_1}, \ldots, \underline{n_l}))$ is bounded by a monotone polynomial in the lengths of the inputs $m_1, \ldots, m_k, n_1, \ldots, n_l$, because

$$\begin{aligned}
&\mathrm{J}\Big(f(\underline{m_1}, \ldots, \underline{m_k}; \underline{n_1}, \ldots, \underline{n_l})\Big) \\
= \;& \mathrm{J}_0(f)\big(|\Phi(\underline{\vec{m}})|\big) \cdot \Big(\sum_{d=1}^{k} \mathrm{J}(\underline{m_d}) + \max_{1 \leq d \leq l} |\Phi(\underline{n_d})| + 1\Big) + \sum_{d=1}^{l} \mathrm{J}(\underline{n_d}) + lh(f) \\
\leq \;& \Big(\mathrm{J}_0(f)\big(|\vec{m}|\big) + 1\Big) \cdot \Big(\sum_{d=1}^{k}(|m_d| + 1) + \sum_{d=1}^{l}(|n_d| + 1) + lh(f)\Big).
\end{aligned}$$

Theorem 4.3 *R'_B is terminating and for every $f \in B$ the derivation length function $\mathcal{D}_{R'_B, f}$ is bounded by a monotone polynomial in the lengths of the inputs.*

The same argumentation as in section 3 yields that this theorem is an immediate consequence of

Lemma 4.4 *If $s, t \in G(B)$ and $s \rightarrow_{R'_B} t$ then $\mathrm{J}(s) > \mathrm{J}(t)$.*

Proof. It is easy to check that $\mathrm{J}(f(\ldots s \ldots)) > \mathrm{J}(f(\ldots t \ldots))$ provided that $\mathrm{J}(s) > \mathrm{J}(t)$. So we only have to prove that $\mathrm{J}(s\sigma) > \mathrm{J}(t\sigma)$ for all $\langle s, t\rangle \in R'_B$ and all ground substitutions σ. To do this we only consider the nontrivial cases.

1. Let $\tilde{f} = SUB^{k,l}_{k',l'}[f, g_1, \ldots, g_{k'}, h_1, \ldots, h_{l'}]$. We consider the case

$$\tilde{f}(s_1, \ldots, s_k; \underline{n_1}, \ldots, \underline{n_l}) \rightarrow_{R'_B} f\big(g_1(\vec{s};), \ldots, g_{k'}(\vec{s};); h_1(\vec{s}; \underline{\vec{n}}), \ldots, h_{l'}(\vec{s}; \underline{\vec{n}})\big).$$

Using $\mathrm{J}(\underline{n}) = |\Phi(\underline{n})| + 1$ for every numeral $\underline{n}$ we compute

$$
\begin{aligned}
&\mathrm{J}\Big(f\big(g_1(\vec{s};),\ldots,g_{k'}(\vec{s};);h_1(\vec{s};\underline{\vec{n}}),\ldots,h_{l'}(\vec{s};\underline{\vec{n}})\big)\Big)\\
=\;& \mathrm{J}_0(f)\Big(|\Phi(g_1(\vec{s};))|,\ldots,|\Phi(g_{k'}(\vec{s};))|\Big)\\
&\cdot\Big(\sum_{d=1}^{k'}\mathrm{J}(g_d(\vec{s};)) + \max_{1\le d\le l'}|\Phi(h_d(\vec{s};\underline{\vec{n}}))| + 1\Big) + \sum_{d=1}^{l'}\mathrm{J}(h_d(\vec{s};\underline{\vec{n}})) + lh(f)\\
\le\;& \Big(\mathrm{J}_0(f)(q_{g_1}(|\Phi(\vec{s})|),\ldots,q_{g_{k'}}(|\Phi(\vec{s})|)\Big)\cdot\Big(\sum_{d=1}^{k'}\Big[\mathrm{J}_0(g_d)(|\Phi(\vec{s})|)\\
&\quad\cdot\Big(\sum_{d=1}^{k}\mathrm{J}(s_d)+1\Big) + lh(g_d)\Big] + \max_{1\le d\le l'}\Big[q_{h_d}(|\Phi(\vec{s})|) + \max_{1\le d\le l}|\Phi(\underline{n}_d)|\Big] + 1\Big)\\
&+\sum_{d=1}^{l'}\Big(\mathrm{J}_0(h_d)(|\Phi(\vec{s})|)\cdot\Big[\sum_{d=1}^{k}\mathrm{J}(s_d) + \max_{1\le d\le l}|\Phi(\underline{n}_d)| + 1\Big]\\
&\quad+\sum_{d=1}^{l}\mathrm{J}(\underline{n}_d) + lh(h_d)\Big) + lh(f)\\
\le\;& \Big[\Big(\mathrm{J}_0(f)(q_{g_1}(|\Phi(\vec{s})|),\ldots,q_{g_{k'}}(|\Phi(\vec{s})|))\Big)\cdot\Big(\sum_{d=1}^{k'}[\mathrm{J}_0(g_d)(|\Phi(\vec{s})|) + lh(g_d)]\\
&\quad+\sum_{d=1}^{l'}q_{h_d}(|\Phi(\vec{s})|) + 1\Big) + \sum_{d=1}^{l'}\mathrm{J}_0(h_d)(|\Phi(\vec{s})|)\Big]\\
&\cdot\Big(\sum_{d=1}^{k}\mathrm{J}(s_d) + \max_{1\le d\le l}|\Phi(\underline{n}_d)| + 1\Big)\\
&\quad+ l\cdot l'\cdot(\max_{1\le d\le l}|\Phi(\underline{n}_d)| + 1) + \sum_{d=1}^{l'}lh(h_d) + lh(f)\\
<\;& \mathrm{J}_0\big(\tilde{f}\big)(|\Phi(\vec{s})|)\cdot\Big(\sum_{d=1}^{k}\mathrm{J}(s_d) + \max_{1\le d\le l}|\Phi(\underline{n}_d)| + 1\Big) + lh\big(\tilde{f}\big)\\
=\;& \mathrm{J}\Big(\tilde{f}(\vec{s};\underline{\vec{n}})\Big)
\end{aligned}
$$

2. Let $f = PREC^{k+1,l}[g,h_1,h_2]$ and consider the case

$$
\begin{aligned}
&f(S_i(;s),s_1,\ldots,s_k;\underline{n_1},\ldots,\underline{n_l})\\
&\quad\rightarrow_{R'_B}\quad h_i\Big(s,s_1,\ldots,s_k;\underline{n_1},\ldots,\underline{n_l},f(s,s_1,\ldots,s_k;\underline{n_1},\ldots,\underline{n_l})\Big)
\end{aligned}
$$

for $i \in \{1,2\}$. Again using $\mathrm{J}(\underline{n}) = |\Phi(\underline{n})| + 1$ for every numeral $\underline{n}$ we compute

$$
\begin{aligned}
&\mathrm{J}\Big(h_i\Big(s, s_1, \ldots, s_k; \underline{n_1}, \ldots, \underline{n_l}, f(s, s_1, \ldots, s_k; \underline{n_1}, \ldots, \underline{n_l})\Big)\Big) \\
= \quad & \mathrm{J}_0(h_i)(|\Phi(s,\vec{s})|) \cdot \Big(\mathrm{J}(s) + \sum_{d=1}^{k} \mathrm{J}(s_d) + \max\{|\Phi(\underline{\vec{n}})|, |\Phi(f(s,\vec{s};\underline{\vec{n}}))|\} + 1\Big) \\
& + \sum_{d=1}^{l} \mathrm{J}(\underline{n}_d) + \mathrm{J}\Big(f(s,\vec{s};\underline{\vec{n}})\Big) + lh(h_i) \\
\leq \quad & \mathrm{J}_0(h_i)(|\Phi(s,\vec{s})|) \cdot \Big(\mathrm{J}(s) + \sum_{d=1}^{k} \mathrm{J}(s_d) + q_f(|\Phi(s,\vec{s})|) + \max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1\Big) \\
& + \mathrm{J}_0(f)(|\Phi(s,\vec{s})|) \cdot \Big(\mathrm{J}(s) + \sum_{d=1}^{k} \mathrm{J}(s_d) + \max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1\Big) \\
& + \sum_{d=1}^{l} \mathrm{J}(\underline{n}_d) + lh(f) + l \cdot (\max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1) + lh(h_i) \\
\leq \quad & \Big[\mathrm{J}_0(h_i)(|\Phi(s,\vec{s})|) \cdot (1 + q_f(|\Phi(s,\vec{s})|)) + l + lh(h_i) \\
& + |\Phi(s)| \cdot \Big([\mathrm{J}_0(h_1)(|\Phi(s,\vec{s})|) + \mathrm{J}_0(h_2)(|\Phi(s,\vec{s})|)] \cdot q_f(|\Phi(s,\vec{s})|) \\
& \quad + l + \max\{lh(h_1), lh(h_2)\}\Big) + \mathrm{J}_0(g)(|\Phi(\vec{s})|)\Big] \\
& \cdot \Big(\mathrm{J}(s) + \sum_{d=1}^{k} \mathrm{J}(s_d) + \max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1\Big) + \sum_{d=1}^{l} \mathrm{J}(\underline{n}_d) + lh(f) \\
\leq \quad & \Big[|\Phi(S_i(;s))| \cdot \Big([\mathrm{J}_0(h_1)(|\Phi(s,\vec{s})|) + \mathrm{J}_0(h_2)(|\Phi(s,\vec{s})|)] \cdot q_f(|\Phi(S_i(s),\vec{s})|) \\
& \quad + l + \max\{lh(h_1), lh(h_2)\}\Big) + \mathrm{J}_0(g)(|\Phi(\vec{s})|)\Big] \\
& \cdot \Big(\mathrm{J}(s) + \sum_{d=1}^{k} \mathrm{J}(s_d) + \max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1\Big) + \sum_{d=1}^{l} \mathrm{J}(\underline{n}_d) + lh(f) \\
< \quad & \mathrm{J}_0(f)\Big(|\Phi(S_i(;s),\vec{s})|\Big) \cdot \Big(\mathrm{J}(S_i(;s)) + \sum_{d=1}^{k} \mathrm{J}(s_d) + \max_{1\leq d\leq l} |\Phi(\underline{n}_d)| + 1\Big) \\
& + \sum_{d=1}^{l} \mathrm{J}(\underline{n}_d) + lh(f) \\
= \quad & \mathrm{J}\Big(f(S_i(;s),\vec{s};\underline{\vec{n}})\Big)
\end{aligned}
$$

□

Corollary 4.5 *Let $k \in \mathbb{N}$ and $f \in B^{k,0}$. Any reasonable R'_B–head reduction* *strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time. Any reasonable R'_B–inside first reduction strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time. (More generally, any reasonable R'_B reduction strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time.)*

Proof. By folklore, there exists a polytime algorithm which, if possible, rewrites any ground term $s \in G(B)$ via an R'_B–head reduction step into its reduct. Consider any deterministic head reduction sequence $f(\underline{m_1}, \ldots, \underline{m_k}) = t_1 \rightarrow_{R'_B} \cdots \rightarrow_{R'_B} t_n = \underline{\Phi(f)(m_1, \ldots, m_k)}$. Then, by Theorem 4.3 and the remark after Definition 4.2, n and the lengths of any t_i for $i \in \{1, \ldots, n\}$ are bounded by a polynomial in the dyadic lengths of $m_1, \ldots, m_k$. Putting things together, the assertion follows. □

5 A non trivial closure property of $\mathcal{POLYTIME}$

In this section we give a non trivial application of the theory developed in the previous sections. We reprove Bellantoni's result stating that $\mathcal{B}$ is closed under predicative recursion with parameter substitution. For simplicity we restrict ourselves to the case of one parameter function.

Theorem 5.1 *Assume that $\mathcal{G} \in \mathcal{B}^{1,1}$, $\mathcal{H}_i \in \mathcal{B}^{2,2}$ and $\mathcal{P} \in \mathcal{B}^{2,1}$. Then there exists a unique polytime function $\mathcal{F} : \mathbb{N}^{2,1} \rightarrow \mathbb{N}$ so that*
$\mathcal{F}(0, m_1; n) = \mathcal{G}(m_1; n)$
$\mathcal{F}(\mathcal{S}_i(m), m_1; n) = \mathcal{H}_i(m, m_1; n, \mathcal{F}(m, m_1; n, \mathcal{P}(m, m_1; n)))$.

The corresponding rewrite system is defined as follows.

Definition 5.2 *Assume that $g \in B^{1,1}$, $h_i \in B^{2,2}$ for $i \in \{1, 2\}$ and $p \in B^{2,1}$. Let f be a new function symbol of arity $2, 1$ and length 1.*
Then $R_{BPS}(g, h_1, h_2, p)$ consists of R'_B plus the rules:
$f(0, x_1; \underline{n}) \rightarrow g(x_1; \underline{n})$
$f(S_i(; x), x_1; \underline{n}) \rightarrow h_i(x, x_1; \underline{n}, f(x, x_1; \underline{n}, p(x, x_1; \underline{n})))$ *for* $i \in \{1, 2\}$.

Using the lexicographic path ordering it can easily be shown that $R_{BPS}(g, h_1, h_2, p)$ is terminating. An alternative and more informative termination proof is given by the following theorem.

Theorem 5.3 *Let R be the rewrite systems given in Definition 5.2. Let f be the corresponding function symbol. Then $\mathcal{D}_{R,f}$ and*

$$\max\{lh(t) : f(\underline{m_1}, \underline{m_2}; \underline{n}) \rightarrow^*_{R_{BPS}} t\}$$

are bounded by a polynomial in the lengths of the inputs m_1, m_2, n.

Proof. We expand the number-theoretic interpretation J_0 from section 4 respecting the new function symbol f. Following [4] we define an appropiate monotone polynomial q_f for majorizing the normal arguments of f:

$$q_f(m_1, m_2) = m_1 \cdot \Big(q_{h_1}(m_1, m_2) + q_{h_2}(m_1, m_2) + q_p(m_1, m_2) + 1\Big) + q_g(m_1, m_2).$$

Again the polymax bounding Lemma 3.2 holds.

In addition to the clauses of the Definition 4.2 we define:

$$\begin{aligned}
&\mathrm{J}_f(m_1, m_2)\\
&\quad = \quad m_1 \cdot \Big([\mathrm{J}_0(h_1)(m_1, m_2) + \mathrm{J}_0(h_2)(m_1, m_2)] \cdot q_f(m_1, m_2)\\
&\qquad\qquad + 2 + \max\{lh(h_1), lh(h_2)\} + lh(p) + \mathrm{J}_0(p)(m_1, m_2)\Big) + \mathrm{J}_0(g)(m_2)\\
&\mathrm{J}_0(f)(m_1, m_2)\\
&\quad = \quad \mathrm{J}_f(m_1, m_2) \cdot \Big(1 + m_1 \cdot q_p(m_1, m_2)\Big)
\end{aligned}$$

Then the interpretation $\mathrm{J}(t)$ of terms $t \in T(B)$ which are not of the form $f(s_1, s_2; t_1)$ is defined in the usual way. In the new case we define and observe

$$\begin{aligned}
&\mathrm{J}\big(f(s_1, s_2; t)\big)\\
&\quad = \quad \mathrm{J}_f(|\Phi(s_1)|, |\Phi(s_2)|) \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + |\Phi(t)| + 1\\
&\qquad\qquad + |\Phi(s_1)| \cdot q_p(|\Phi(s_1)|, |\Phi(s_2)|)\Big) + \mathrm{J}(t) + 1\\
&\quad \leq \quad \mathrm{J}_0(f)(|\Phi(s_1)|, |\Phi(s_2)|) \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + |\Phi(t)| + 1\Big) + \mathrm{J}(t) + lh(f)
\end{aligned}$$

Therefore all observations from section 4 transfer to the new system, e.g. $\mathrm{J}(\underline{m}) = |m| + 1$ and $\mathrm{J}(t) \geq lh(t)$. Also we see that $\mathrm{J}(f(\underline{m_1}, \underline{m_2}; \underline{n}))$ is bounded by a monotone polynomial in the lengths of the inputs m_1, m_2, n. Again it is easy to check that $\mathrm{J}(\psi(\ldots s \ldots)) > \mathrm{J}(\psi(\ldots t \ldots))$ provided that $\mathrm{J}(s) > \mathrm{J}(t)$. So we only have to prove that $\mathrm{J}(s\sigma) > \mathrm{J}(t\sigma)$ for all $\langle s, t\rangle \in R_{BPS}$ and all ground substitutions σ. As the estimations in the proof of Lemma 4.4 carry over to the new interpretation we only have to consider the new case:

$$f(S_i(; s_1), s_2; \underline{n}) \to_{R'_B} h_i\big(s_1, s_2; \underline{n}, f(s_1, s_2; p(s_1, s_2; \underline{n}))\big)$$

for $i \in \{1, 2\}$. Using $\mathrm{J}(\underline{n}) = |\Phi(\underline{n})| + 1$ for every numeral $\underline{n}$ we compute

$$\begin{aligned}
&\mathrm{J}\Big(h_i\big(s_1, s_2; \underline{n}, f(s_1, s_2; p(s_1, s_2; \underline{n}))\big)\Big)\\
&\quad \leq \quad \mathrm{J}_0(h_i)(|\Phi(s_1)|, |\Phi(s_2)|) \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + q_f(|\Phi(s_1)|, |\Phi(s_2)|)\\
&\qquad\qquad + q_p(|\Phi(s_1)|, |\Phi(s_2)|) + |\Phi(\underline{n})| + 1\Big) + \mathrm{J}(\underline{n}) + lh(h_i)
\end{aligned}$$

$$
\begin{aligned}
& + \mathrm{J}_f(|\Phi(s_1)|, |\Phi(s_2)|) \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + q_p(|\Phi(s_1)|, |\Phi(s_2)|) + |\Phi(\underline{n})| + 1 \\
& \quad + |\Phi(s_1)| \cdot q_p(|\Phi(s_1)|, |\Phi(s_2)|) \Big) \\
& + \mathrm{J}_0(p)(|\Phi(s_1)|, |\Phi(s_2)|) \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + |\Phi(\underline{n})| + 1 \Big) \\
& \quad + \mathrm{J}(\underline{n}) + lh(p) + 1 \\
\leq \; & \Big[\mathrm{J}_0(h_i)(|\Phi(s_1)|, |\Phi(s_2)|) \cdot (1 + q_f(|\Phi(s_1)|, |\Phi(s_2)|)) + 1 + lh(h_i) + lh(p) \\
& \quad + \mathrm{J}_0(p)(|\Phi(s_1)|, |\Phi(s_2)|) + \mathrm{J}_f(|\Phi(s_1)|, |\Phi(s_2)|) \Big] \\
& \cdot \Big(\mathrm{J}(s_1) + \mathrm{J}(s_2) + |\Phi(\underline{n})| + 1 + (|\Phi(s_1)| + 1) \cdot q_p(|\Phi(s_1)|, |\Phi(s_2)|) \Big) \\
& \quad + \mathrm{J}(\underline{n}) + 1 \\
< \; & \mathrm{J}\Big(f(S_i(; s_1), s_2; \underline{n}) \Big)
\end{aligned}
$$

□

Corollary 5.4 *Any reasonable $R_{BPS}(g, h_1, h_2, p)$–head reduction strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time. Any reasonable $R_{BPS}(g, h_1, h_2, p)$–inside first reduction strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time. (More generally, any reasonable $R_{BPS}(g, h_1, h_2, p)$ reduction strategy yields an algorithm for $\Phi(f)$ which runs in polynomial time.)*

Proof. By folklore, there exists a polytime algorithm which, if possible, rewrites any ground term $s \in G(B)$ via an $R_{BPS}(g, h_1, h_2, p)$–head reduction step into its reduct. Consider any deterministic head reduction sequence $f(\underline{m_1}, \underline{m_2}; \underline{n}) = t_1 \to_{R'_B} \cdots \to_{R'_B} t_k = \underline{\Phi(f)(m_1, m_2; n)}$. Then, by Theorem 5.3 and the remark in the proof of Theorem 5.3, k and the length of any t_i for $i \in \{1, \ldots, k\}$ are bounded by a polynomial in the dyadic lengths of m_1, m_2, n. Putting things together, the assertion follows. □

Possible extensions: The term rewriting approach based on derivation lengths classifications can easily be adapted for showing that predicative recursion is also closed - even with respect to derivation lengths - under simultaneous predicative recursion.

Using derivation lengths classifications of appropriate term graph rewriting systems or appropriate logic programs the authors were able to show that predicative recursion is closed - with respect to derivation or computation lengths - under predicative course of values recursion and more generally under predicative descent recursion [2].

The distinction between normal and safe arguments can be used to define fragments of bounded predicative arithmetic which are related to the well-known fragments of bounded arithmetic S_2^i and T_2^i introduced in [5]. In [1] the first author was recently able to seperate these predicative fragments by adapting methods of ordinal analysis for fragments of PA [12].

Acknowledgements: The second author would like to thank E.A. Cichon and Pierre Lescanne for their hospitality during his stay at Nancy as a member of the Eureca group from 1/10/94 - 31/3/95 under HCM-grant 293F0740051507126. He would like to thank S.S. Wainer for his hospitality while visiting Leeds University under this grant from 19/3/95-25/3/95. The possibility of a rewrite analysis of the polytime functions was suggested to the second author in stimulating discussions on proof and recursion theory by Stan Wainer during the second author's stay in Leeds. (The results of this paper have been presented at the workshop on Termination (Nancy 1995) and the Amsterdam Münster Intercity seminar (Amsterdam 1995).) The authors would like to thank the referees for their detailed and helpful comments.

References

[1] Beckmann, A.: *Seperating fragments of bounded predicative arithmetic*. Dissertation (1996).

[2] Beckmann, A. and Weiermann, A.: *Investigations on subrecursion via logic programming*. Preprint, Münster (1995).

[3] Bellantoni, S. and Cook, S.: *A new recursion-theoretic characterization of the polytime functions.* Comput. Complexity 2, No. 2 (1992), 97-110.

[4] Bellantoni, S.: *Predicative recursion and computational complexity.* Dissertation, Toronto (1992).

[5] Buss, S.: *Bounded arithmetic*. Bibliopolis 1986.

[6] Cichon, E.A. and Weiermann, A.: *Term rewriting theory for the primitive recursive functions.* To appear in: Ann. Pure Appl. Logic.

[7] Dershowitz, N. & Jouannaud, J.P.: *Rewrite systems.* In: Handbook of Theoretical Computer Science B: Formal Methods and Semantics, J. van Leeuwen (ed.) North-Holland 1990, pp. 243-320.

[8] Handley,W.G. and Wainer, S.S.: *Equational derivation vs. computation.* Ann. Pure Appl. Logic 70 (1994), 17-49.

[9] Hofbauer, D.: *Termination proofs by multiset path orderings imply primitive recursive derivation lengths.* Proc. 2nd ALP. Lecture Notes in Computer Science 463 (1990), 347-358.

[10] Leivant, D.: *Predicative recurrence in finite type.* In A. Nerode and Y.V. Matiyasevich (eds.), Logical Foundations of Computer Science. Springer Lecture Notes in Computer Science 813 (1994), 227-239.

[11] Möllerfeld, M. and Weiermann, A.: *A uniform approach to $\prec$-recursion.* Preprint, Münster (1995) (submitted).

[12] Pohlers, W.: *Proof theory: An Introduction.* Springer Lecture Notes in Mathematics 1407, 1989.

[13] Simmons, H.: *The realm of primitive recursion.* Arch. Math. Logic 27 (1988), 177-188.

[14] Weiermann, A.: *Termination proofs for term rewriting systems by lexicographic path orderings yield multiply recursive derivation lengths.* Theoretical Computer Science 139 (1995), 355-362.

[15] Weiermann, A.: *Rewriting theory for the Hydra battle and the extended Grzegorczyk hierarchy*, Preprint, Nancy and Münster (1995) (submitted).

[16] Weiermann, A.: *A proof of strongly uniform termination for Gödel's T by methods from local predicativity.* Preprint, Münster (1995) (submitted).