Polynomial Interpretations and the Complexity of Algorithms

E.A.Cichon and Pierre Lescanne

Abstract

The ability to use a polynomial iterpretation to prove termination of a rewrite system naturally prompts the question as to what restriction on complexity this imposes. The main result of this paper is that a polynomial interpretation termination proof of a rewrite system $\mathcal{R}$ which computes a number theoretic function implies a polynomial bound on that function's rate of growth.

1 Introduction

The ability to use a polynomial iterpretation to prove termination of a rewrite system, a method due to Lankford [4, 5], naturally prompts the question as to what restriction on complexity this imposes. Various claims and conjectures have been made, significant amongst which is:

> *Of course, polynomial interpretations do not suffice in general, since they give a polynomial upper bound on the complexity of the computations by $\mathcal{R}$, interpreted as a program computing over integers, whereas arbitrary recursive functions can be defined by term rewriting systems.*
>
> HUET AND OPPEN, 1980

At first sight, it would appear that this claim, which appears in [3], has, once and for all, been refuted by Lautemann [6] and Geupel [2], who show that polynomial interpretation admits doubly exponential derivation lengths. Lautemann and Geupel give the following system of rewrites for the number theoretic squaring function, together with a polynomial interpretation which

proves termination:

Rewrite Rules			*Polynomial Interpretation*		
$x + 0$	$\rightarrow$	x	$[0]$	$=$	2
$x + s(y)$	$\rightarrow$	$s(x + y)$	$[s](X)$	$=$	$X + 1$
$d(0)$	$\rightarrow$	0	$[+](X, Y)$	$=$	$X + 2Y$
$d(s(x))$	$\rightarrow$	$s(s(d(x)))$	$[d](X)$	$=$	$3X$
$q(0)$	$\rightarrow$	0	$[q](X)$	$=$	X^3
$q(s(x))$	$\rightarrow$	$s(q(x) + d(x))$			

They then consider terms of the form $q^{(n)}(s(s(0)))$. These terms satisfy:

$$q^{(n)}(s(s(0))) \ \rightarrow^*_{\mathcal{R}} s^{2^{2^n}}(0)$$

Since no rule increases the number of occurences of s by more than one, the length of this derivation must be doubly exponential. Lautemann and Geupel then go on to prove:

Theorem(Geupel [2],Lautemann [6])
If $\mathcal{R}$ has a polynomial interpretation then there is a constant c such that, for any terms $t,\ s$,

$$\text{if } t \rightarrow^n_{\mathcal{R}} s \text{ then } n \leq 2^{2^{c.\mathcal{L}(t)}}$$

where $\mathcal{L}(t)$ is the number of symbols in t and $t \rightarrow^n_{\mathcal{R}} s$ means that t rewrites to s in n steps under the rules in $\mathcal{R}$.

There is, however, more than one notion of complexity which can be employed. By *computational complexity* we shall mean the assignment of a measure to rewrite systems in terms of lengths of derivations. *Algorithmic complexity* will refer to the classification of a rewrite system according to the *form* of its rules. To illustrate this with an example, consider a function defined by primitive recursive schemes. This means that the function is defined by a set of equations, all of which are formed according to the criteria set out in the definition of the Primitive Recursive functions. As pointed out by Plaisted in [7], such equations are easily oriented into a set of rewrite rules which compute the function, and so these rules form a rewrite system with primitive recursive algorithmic complexity.

Thus we are led to considering the problem of classifying the algorithmic complexity of rewrite systems whose termination is provable by polynomial interpretation. We shall restrict ourselves to rewrite systems which compute number-theoretic functions. The main result of this paper is then that a polynomial interpretation termination proof of a rewrite system $\mathcal{R}$ which computes a number theoretic function implies a polynomial bound on that function's rate of growth.

As a corollary, one can obtain the correctness of Huet and Oppen's conjecture.

2 Preliminaries

The term rewriting notations used in this paper will be largely based on those advocated by N.Dershowitz and J-P.Jouannaud in [1]. This justifies our omission of the customary paragraphs dealing with these issues.

2.1 The {0,s}-Constructor Discipline

The main emphasis in this paper is on the use of term rewriting systems for defining *number theoretic functions*, that is, functions:$\mathbf{N}^k \mapsto \mathbf{N}$. We therefore restrict our attention to term rewriting systems which are assumed to have the following properties:

1. $\mathcal{R}$ is a finite set of rewrite rules over the set $\mathcal{T}(\mathcal{F}, \mathcal{X})$ of terms.
2. The set $\mathcal{F}$ of function symbols is finite, contains only one constant (0-ary function symbol), 0, and contains the unary function symbol s.
3. $\mathcal{R}$ is terminating and confluent, and the set

$$\{0, s(0), \ldots, \underbrace{s(s(\ldots s(s}_{n \text{ times}}(0))\ldots)), \ldots\}$$

   is precisely the set of normal forms for the ground terms in $\mathcal{T}$. This means that for every function symbol f in $\mathcal{F}$, which is not 0 or s, there is an f-eliminating rule in $\mathcal{R}$.

Henceforth we write $s^{(n)}(0)$, and frequently $s^n 0$, for $\underbrace{s(s(\ldots s(s}_{n \text{ times}}(0))\ldots))$.

The term $s^n 0$ represents the numeral n. A function symbol f in $\mathcal{F}$ represents the number theoretic function $\{f\}$ if

$$f(s^{m_1}0, \ldots, s^{m_k}0) \rightarrow^!_{\mathcal{R}} s^m 0$$
$$\text{if and only if}$$
$$\{f\}(m_1, \ldots, m_k) = m.$$

It is in this sense, therefore, that we say $\mathcal{R}$ *computes* $\{f\}$.

2.2 Polynomial Interpretation Termination Proofs

A *polynomial interpretation termination proof* for a rewrite system $\mathcal{R}$ over a set $\mathcal{T}(\mathcal{F}, \mathcal{X})$ of terms is achieved by assigning to each function symbol f in $\mathcal{F}$ a polynomial with integer coefficients. We denote this polynomial by $[f]$. If f is n-ary then $[f]$ must be a polynomial in n variables. The polynomial $[f]$ must satisfy the monotonicity condition:

$$x < y \Longrightarrow [f](\cdots x \cdots) < [f](\cdots y \cdots)$$

and must ensure that terms are mapped into nonnegative integers only. Each rule must be reducing with respect to the its interpretation, that is

$$l \to r \in \mathcal{R} \Longrightarrow [r] < [l]$$

for all values of variables greater than the minimum of the interpretations of the ground terms.

Remark 1 *Whenever $\mathcal{R}$ has a polynomial interpretation termination proof the following useful inequality arises:*

$$\textit{if } t \to^*_{\mathcal{R}} s \textit{ then } [s] < [t].$$

3 Exponential Functions and Polynomial Interpretations

In this section we shall show that functions of exponential growth cannot be computed by rewrite systems with polynomial interpretation termination proofs. The proof is achieved by first establishing a relationship between the rate of growth of a function and the length of its computation and then using Lautemann and Geupel theorem.

Definition 1 *We define the height, $|t|$, of a term t in $T(F,X)$ as follows:*

$$|t| = \begin{cases} 0 & \textit{when } t \textit{ is a constant or a variable,} \\ \max_{i \in 1..n}\{|t_i|\} + 1 & \textit{when } t = f_k(\vec{t}). \end{cases}$$

where $\vec{t} = t_1, \ldots, t_n$.

Definition 2 *Suppose that $\mathcal{R}$ is a finite set of rewrite rules $\{l(\vec{x}) \to r(\vec{x})\}$. Let*

$$M_{\mathcal{R}} = \max_{l(\vec{x}) \to r(\vec{x}) \in \mathcal{R}} \{|r(\vec{x})|\}$$

Theorem 1

$$\textit{If } t \to^1_{\mathcal{R}} s \textit{ then } |s| \dot{-} |t| \leq M_{\mathcal{R}}.$$

and hence

$$\textit{If } t \to^n_{\mathcal{R}} s \textit{ then } n \geq \frac{|s| \dot{-} |t|}{M_{\mathcal{R}}}$$

Proof: *The proof is given by lemmas 1, 2, 3 below.* □

Lemma 1 *For any substitution* $\vec{d}$ $(= d_1, \ldots, d_n)$ *for* $\vec{x}$ $(= x_1, \ldots, x_n)$,

$$|\, t(\vec{d})\,| \leq |\, t(\vec{x})\,| + \max_{i \in 1..n}\{|\, d_i\,|\}.$$

Proof: *The proof is by induction on the term tree for* $t(\vec{x})$. *When* t *is a constant or a variable, the result is trivial.*
If $t(\vec{x}) = f(t_1(\vec{x}), \ldots, t_m(\vec{x}))$ *then*

$$\begin{aligned}
|\, t(\vec{d})\,| &= \max_{j \in 1..m}\{|\, t_j(\vec{d})\,|\} + 1 \\
&\leq \max_{j \in 1..m}\{|\, t_j(\vec{x})\,| + \max_{i \in 1..n}\{|\, d_i\,|\}\} + 1 \\
&\qquad\qquad \textit{(by induction hypothesis)} \\
&\leq \max_{j \in 1..m}\{|\, t_j(\vec{x})\,|\} + 1 + \max_{i \in 1..n}\{|\, d_i\,|\} \\
&= |\, t(\vec{x})\,| + \max_{i \in 1..n}\{|\, d_i\,|\}.
\end{aligned}$$

□

Lemma 2 *For any substitution* $\vec{d}$ *for* $\vec{x}$,

$$|\, r_i(\vec{d})\,| \leq |\, l_i(\vec{d})\,| + M_{\mathcal{R}}$$

Proof: *By lemma 1,*

$$\begin{aligned}
|\, r_i(\vec{d})\,| &\leq |\, r_i(\vec{x})\,| + \max_{i \in 1..n}\{|\, d_i\,|\} \\
&\leq M_{\mathcal{R}} + \max_{i \in 1..n}\{|\, d_i\,|\} \\
&\leq M_{\mathcal{R}} + |\, l_i(\vec{d})\,|\,.
\end{aligned}$$

□

The next lemma generalises lemma 2 to the case where a one step rewrite occurs by applying a rule in $\mathcal{R}$ to a proper subterm of a term t. We shall use the notation $t[u]$ to denote a term t with u as a proper subterm.

Lemma 3 *For any substitution* $\vec{d}$ *for* $\vec{x}$,

$$|\, t[r_i(\vec{d})]\,| \dot{-} |\, t[l_i(\vec{d})]\,| \leq M_{\mathcal{R}}.$$

Proof: *This is now a straightforward induction over the term tree for* t. □

We can apply the results above to obtain a connection between algorithmic and computational complexity of a function $\{f\}$ computed by a rewrite system $\mathcal{R}$. The computation of $\{f\}(m_1, \ldots, m_k)$ is achieved by normalising $f(s^{m_1}(0), \ldots, s^{m_k}(0))$ under the rules in $\mathcal{R}$.

Suppose that $\{f\}(m_1,\ldots,m_k) = m$ so that

$$f(s^{m_1}(0),\ldots,s^{m_k}(0)) \rightarrow^{!} s^{m}(0).$$

By theorem 1, if n is the number of steps to normalisation, we obtain

$$n \;\geq\; \frac{|\,s^{m}(0)\,| \mathbin{\dot{-}} |\,f(s^{m_1}(0),\ldots,s^{m_k}(0))\,|}{M_{\mathcal{R}}}$$

Since

$$|\,s^{m}(0)\,| = m = \{f\}(m_1,\ldots,m_k)$$

and

$$|\,f(s^{m_1}(0),\ldots,s^{m_k}(0))\,| = \max_{i\in 1..k}\{m_i\} + 1$$

we therefore have

$$n \;\geq\; \frac{\{f\}(m_1,\ldots,m_k) \mathbin{\dot{-}} \max_{i\in 1..k}\{m_i\} + 1}{M_{\mathcal{R}}}$$

In particular, if $\{f\}$ is an exponential function, then the number of steps to normalisation is at least exponential in the height of the starting term.

We now obtain:

Theorem 2 *If $\mathcal{R}$ computes the number theoretic function $\{f\}$ and $\{f\}$ has exponential growth then $\mathcal{R}$ has no polynomial interpretation termination proof.*
Proof: *Without serious loss of generality and with considerably greater clarity we give the proof in the case where*

$$\{f\}(x) = 2^{x}.$$

For any j, consider the term $f^{(j)}(0)$. Then

$$f^{(j)}(0) \rightarrow^{n}_{\mathcal{R}} s^{m}(0).$$

We write 2_k to denote $2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}\Big\}k$ times. We have

$$m = 2_{j-1}$$

and, by theorem 1,

$$n \geq \frac{2_{j-1} - j + 1}{M_{\mathcal{R}}}.$$

Now, if R had a polynomial interpretation termination proof then by Lautemann and Geupel theorem we would have

$$\begin{aligned} n &\leq 2^{2^{c\mathcal{L}(f^{(j)}(0))}} \\ &= 2^{2^{cj}} \end{aligned}$$

so that

$$\frac{2_{j-1} - j + 1}{M_{\mathcal{R}}} \leq 2^{2^{cj}}, \textit{for all } j$$

which is clearly impossible. □

Theorem 2 shows that functions with at least exponential growth cannot be computed by rewrite systems with polynomial interpretation termination proof.

In the next section we shall sharpen this and show that a polynomial interpretation termination proof of a rewrite system $\mathcal{R}$ which computes a number theoretic function implies a polynomial bound on that function's rate of growth.

4 Polynomial Interpretation Termination Proof Implies A Polynomial Bound On $\{f\}$

This section is devoted to the proof of the following theorem:

Theorem 3 *Suppose that $\mathcal{R}$ is a rewrite system which computes the function $\{f\}$ and that $\mathcal{R}$ has a polynomial interpretation termination proof. Then, according to the interpretation of s, we have the following:*

1. *If $[s](X) = X + q$, where q is a constant ≥ 1, then $\{f\}$ is bounded by a polynomial function P i.e. $\{f\}(\vec{x}) \leq P(\vec{x})$, for all $\vec{x}$.*
2. *If $[s](X) = aX^p + Q(X)$, where $d^oQ < p$ and either $a \geq 1$ and $p > 1$ or $a > 1$ and $p \geq 1$, then $\{f\}$ is bounded by a linear polynomial.*

4.1 When the constructor s has a linear interpretation

In this section we consider the case when the constructor s has a linear interpretation, namely

$$[s](X) = aX + q \text{ where } a(\geq 1) \text{ and } q(\geq 1) \text{ are constants.}$$

Without loss of generality, we suppose that f is a monadic function. Assume that $[f](X)$ is a polynomial of degree d with leading coefficient c and $[0] = b$.

There are two cases to consider:

Case 1 $a = 1$.

We have

$$[s^n(0)] = b + nq$$

Therefore

$$[f(s^n(0))] = [f](b + nq)$$

Since $[f]$ is a polynomial, so clearly is $\lambda n.[f](b + nq)$. ¿From the inequality

$$[f(s^n(0))] > [s^{\{f\}(n)}(0)]$$

which is a consequence of remark 1 and $f(s^n(0)) \rightarrow_{\mathcal{R}}^* s^{\{f\}(n)}(0)$, it follows that $\{f\}$ is polynomially bounded.

Case 2 $a > 1$.

We now have

$$[s^n(0)] = ba^n + R(a) \text{ with } d^oR < n$$

Therefore

$$[f(s^n(0))] = [f](ba^n + R(a)) \sim cb^d a^{nd}$$

On the other hand

$$[s^{\{f\}(n)}(0)] \sim ba^{\{f\}(n)}$$

¿From the inequality

$$[f(s^n(0))] > [s^{\{f\}(n)}(0)]$$

one gets the following inequality on the leading terms:

$$cb^d a^{nd} \geq ba^{\{f\}(n)}$$

which implies

$$n \geq \frac{1}{d}\{f\}(n) + constant$$

This means that in this case the function $\{f\}$ is bounded by a linear polynomial.

An example of such a situation is the following rewrite system which has exponential derivation lengths:

$$\begin{array}{rcl} f(0,y) & \rightarrow & y \\ f(s(x),y) & \rightarrow & f(x,f(x,y)) \end{array}$$

Its termination can be proved by the polynomial interpretation

$$\begin{array}{rcl} [0] & = & 2 \\ [s](X) & = & 2X+1 \\ [f](X,Y) & = & X+Y \end{array}$$

and indeed $\{f\}(m,n) = n$.

4.2 When the constructor s has a non linear interpretation

In this section we consider the case when the constructor s has a non linear interpretation, namely

$$[s](X) = aX^p + Q(X) \text{ with } p \geq 2 \text{ and } d^oQ < p$$

Again, we suppose that f is a monadic function and that $[f](X)$ is a polynomial of degree d with leading coefficient c and $[0] = b$. One has

$$[s^n(0)] = a^n b^{p^n} + R(b) \text{ with } d^o R < p^n$$

Therefore

$$[f(s^n(0))] = [f](a^n b^{p^n} + R(b)) \sim c a^{dn} b^{dp^n}$$

On the other hand

$$[s^{\{f\}(n)}(0)] \sim a^{\{f\}(n)} b^{p^{\{f\}(n)}}$$

Once more, from the inequality

$$[f(s^n(0))] > [s^{\{f\}(n)}(0)]$$

we get the following inequality on the leading terms:

$$c a^{dn} b^{dp^n} \geq a^{\{f\}(n)} b^{p^{\{f\}(n)}}$$

which implies

$$n + log_p d \geq \{f\}(n)$$

if one considers only the exponents of b. This means that in this case the function $\{f\}$ is less than linear. A natural example of such a situation is the system:

$$\begin{array}{rcl} half(s(s(x))) & \rightarrow & s(half(x)) \\ half(s(0)) & \rightarrow & 0 \\ half(0) & \rightarrow & 0 \end{array}$$

Its termination can be proved by the polynomial interpretation

$$\begin{array}{rcl} [s](X) & = & X^2 \\ [half](X) & = & X^d \end{array}$$

and indeed $\{half\}(n) = \lfloor \frac{n}{2} \rfloor$ is less than n.

5 Practical issues

This result has interesting practical consequences. First it sets a strict limit on the possible interpretations of the constructors, namely essentially $[s](X) = X + c$ and $[0] = 2$ and no specific limit for the defined functions. On the other hand, we can guess from the function $\{f\}$ whether the rewrite system that defines f has a polynomial interpretation proof of termination. Such a proof exists only if $\{f\}$ has polynomial growth. For instance, a system that defines the factorial or the exponential cannot be proved to terminate using polynomial interpretations.

References

[1] N. Dershowitz and J-P. Jouannaud. Notations for rewriting, 1991. LaTeXscript.

[2] O. Geupel. Terminationbeweise bei termersetzungssytem, 1988. Diplomarbeit.

[3] G. Huet and D.C. Oppen. Equations and rewrite rules: A survey. In R. Book, editor, *Formal Language Theory: Perspectives and Open Problems*, pages 349–405. Academic Press, 1980.

[4] D. Lankford. Canonical algebraic simplification in computational logic. Report ATP–25, University of Texas, 1975.

[5] D. Lankford. On proving term rewriting systems are noetherian. Report MTP-3, Louisiana Tech. University, 1979.

[6] C. Lautemann. A note on polynomial interpretation. In *Bulletin of the European Association for Theoretical Computer Science*, volume 4, pages 129–131, October 1988.

[7] D.A. Plaisted. A recursively defined ordering for proving termination of term rewriting systems. Report R-78-943, Department of Computer Science, University of Illinois, Urbana, Illinois., September 1978.