Automating Elementary Interpretations

Harald Zankl, Sarah Winkler, and Aart Middeldorp

Institute of Computer Science, University of Innsbruck, 6020 Innsbruck, Austria

Abstract

We report on an implementation of elementary interpretations for automatic termination proofs.

1 Introduction

Proving termination of rewrite systems by polynomial interpretations is well-studied. In this work we go beyond polynomials by considering a subset of elementary functions including exponentiation. The approach is motivated by Lescanne's factorial example [3]

$$\begin{array}{ccc} \mathsf{0} + x \to x & \mathsf{0} \cdot x \to \mathsf{0} & \mathsf{fact}(\mathsf{0}) \to \mathsf{s}(\mathsf{0}) \\ \mathsf{s}(x) + y \to \mathsf{s}(x+y) & \mathsf{s}(x) \cdot y \to x \cdot y + y & \mathsf{fact}(\mathsf{s}(x)) \to \mathsf{s}(x) \cdot \mathsf{fact}(x) \\ x \cdot (y+z) \to x \cdot y + x \cdot z & & \end{array}$$

which does not admit a (direct) termination proof by polynomials. Consider the algebra $\mathcal{A}$ with carrier $\mathbb{N}_{\geqslant 1}$ and the interpretation functions $\mathsf{0}_{\mathcal{A}} = 2$, $\mathsf{s}_{\mathcal{A}}(x) = x+2$, $x +_{\mathcal{A}} y = 2x+y+1$, $x \cdot_{\mathcal{A}} y = 2^x y$, and $\mathsf{fact}_{\mathcal{A}}(x) = 2^{2^x}$. They establish termination of the TRS, since for every rule the term interpretation of the left-hand side is larger than that of the right-hand side, i.e., for all $x, y, z \in \mathbb{N}_{\geqslant 1}$:

$$\begin{array}{ccc} x + 5 > x & 2^2 x > 2 & 2^{2^2} > 4 \\ 2x + y + 5 > 2x + y + 3 & 2^{x+2} y > 2^{x+1} y + y + 1 & 2^{2^{x+2}} > 2^{x+2} 2^{2^x} = 2^{x+2+2^x} \\ 2^x(2y + z + 1) > 2^{x+1} y + 2^x z + 1 & & \end{array}$$

In this note we show how to automate the search for such interpretation functions. In particular one has to (a) find suitable coefficients for the interpretations, (b) evaluate the term interpretations, and (c) compare two term interpretations. In the sequel we address these issues in reverse order and also discuss the limitations of this approach.

2 Automation of Elementary Algebras

We assume familiarity with term rewriting [1]. For a set of function symbols $\mathcal{F}$, a (well-founded) algebra $\mathcal{A} = (A, \{f_{\mathcal{A}} \mid f \in \mathcal{F}\}, >)$ consists of a carrier A, (a set of) interpretation functions $f_{\mathcal{A}} : A \times \cdots \times A \to A$, and a well-founded order $>$ on A. An interpretation function $f_{\mathcal{A}}$ is *monotone* if $a > b$ implies $f_{\mathcal{A}}(\ldots, a_{i-1}, a, a_{i+1}, \ldots) > f_{\mathcal{A}}(\ldots, a_{i-1}, b, a_{i+1}, \ldots)$. An algebra is *monotone* if all its interpretation functions are monotone. A TRS $\mathcal{R}$ is *compatible* with an algebra $\mathcal{A}$ if $[\alpha]_{\mathcal{A}}(\ell) > [\alpha]_{\mathcal{A}}(r)$ for every $\ell \to r \in \mathcal{R}$ and assignment α. Here $[\alpha]_{\mathcal{A}}(t)$ denotes the value resulting when interpreting the term t in the algebra $\mathcal{A}$ under the assignment α. A TRS is terminating if and only if it is compatible with a well-founded monotone algebra.

Inspired by the above example we use interpretation functions of the following shape:

▶ **Definition 1.** A *fixed-base elementary interpretation function* (FBI) of depth 0 is a linear function $f(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i f_i + f_0$ and an FBI of depth $d+1$ has the shape

$$f(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i f_i + f_0 + b^{f'(\overline{x})} \left(\sum_{1 \leqslant i \leqslant n} x_i \hat{f}_i + \hat{f}_0 \right) \tag{1}$$

where $f_0, f_1, \ldots, f_n, \hat{f}_0, \hat{f}_1, \ldots, \hat{f}_n$ are natural numbers, $f'(\overline{x})$ is an FBI of depth d, and $b \geqslant 2$ is a fixed natural number. We use the abbreviations $\dot{f}(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i f_i + f_0$ and $\hat{f}(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i \hat{f}_i + \hat{f}_0$. An *FBI algebra* has $\mathbb{N}_{\geqslant 1}$ as carrier and FBIs as interpretation functions for all function symbols in the signature.

We treat an FBI $f(\overline{x})$ of depth 0 as $\sum_{1 \leqslant i \leqslant n} x_i f_i + f_0 + b^0 0$ to avoid case distinctions. Hence in the sequel we will use FBIs $f(\overline{x})$ and $g(\overline{x})$ of the shape (1).

The following recursive definition reduces the comparison of FBIs to the comparison of non-linear polynomials.

▶ **Definition 2.** Let $\lfloor b^{f'(\overline{x})} \rfloor = \big((\dot{f}'(\overline{x}) + \hat{f}'(\overline{x}) = 0) \ ? \ 1 \ : \ b\,(\dot{f}'(\overline{x}) + \hat{f}'(\overline{x}))\big)$. Note that $b^{f(\overline{x})} \geqslant \lfloor b^{f(\overline{x})} \rfloor$. Further let $p(\overline{x}) = \dot{f}'(\overline{x}) + \hat{f}'(\overline{x}) - \dot{g}'(\overline{x}) - \hat{g}'(\overline{x})$ and $h(\overline{x}) = \lfloor b^{p(\overline{x})} \rfloor \hat{f}(\overline{x}) - \hat{g}(\overline{x})$. We define

$$
\begin{aligned}
&[f(\overline{x}) > g(\overline{x})] \;=\; (\hat{g}(\overline{x}) > 0 \rightarrow [f'(\overline{x}) \geqslant g'(\overline{x})]) \;\wedge\; \big(\\
&\quad (\hat{f}(\overline{x}) > 0 \wedge [f'(\overline{x})\, b > g(\overline{x})]) \;\vee && \text{(d)}\\
&\quad (\dot{f}(\overline{x}) \geqslant \dot{g}(\overline{x}) \wedge \hat{f}(\overline{x}) \geqslant \hat{g}(\overline{x}) \wedge \\
&\qquad ((\hat{f}(\overline{x}) > 0 \wedge [f'(\overline{x}) > g'(\overline{x})]) \vee \dot{f}(\overline{x}) > \dot{g}(\overline{x}) \vee \hat{f}(\overline{x}) > \hat{g}(\overline{x}))) \;\vee && \text{(e)}\\
&\quad (h(\overline{x}) \geqslant 0 \wedge p(\overline{x}) \geqslant 0 \wedge \hat{f}'(\overline{x}) \geqslant \hat{g}'(\overline{x}) \wedge \\
&\qquad \dot{f}(\overline{x}) + \lfloor b^{g'(\overline{x})} \rfloor \lfloor b^{p(\overline{x})} \rfloor \hat{f}(\overline{x}) > \dot{g}(\overline{x}) + \lfloor b^{g'(\overline{x})} \rfloor \hat{g}(\overline{x})) \;\big) && \text{(f)}
\end{aligned}
$$

The encodings of comparisons are sound.

▶ **Lemma 3.** *If $[f(\overline{x}) > g(\overline{x})]$ holds then $[\alpha](f(\overline{x})) > [\alpha](g(\overline{x}))$ for all assignments α.*

The following example shows that FBIs are not closed under addition and composition, which complicates the evaluation of a term interpretation.

▶ **Example 4.** The sum $2^x + 2^y$ of the FBIs 2^x and 2^y has no FBI representation. Substituting the FBI $2^y + 1$ for x in the FBI $2^x x$ results in $2^{2^y+1}(2^y + 1) = 2^{2^y+y+1} + 2^{2^y+1}$, which also has no equivalent FBI representation.

We thus define under- and overapproximations for arithmetic operations.

▶ **Definition 5.**

(a) Multiplication of an FBI by a scalar again yields an FBI, i.e.

$$f(\overline{x})\, a = \sum_{1 \leqslant i \leqslant n} x_i f_i a + f_0 a + b^{f'(\overline{x})} \Big(\sum_{1 \leqslant i \leqslant n} x_i \hat{f}_i a + \hat{f}_0 a \Big)$$

(b) For addition, we first introduce $\mathrm{fmin}(f, g)$ and $\mathrm{fmax}(f, g)$ as the coefficient-wise minimum and maximum of FBIs f and g, respectively. These lower (upper) bounds admit the approximations

$$f(\overline{x}) +_\mu g(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i (f_i + g_i) + (f_0 + g_0) + b^{e_\mu(\overline{x})} \Big(\sum_{1 \leqslant i \leqslant n} x_i (\hat{f}_i + \hat{g}_i) + (\hat{f}_0 + \hat{g}_0) \Big)$$

$$f(\overline{x}) +_\nu g(\overline{x}) = \sum_{1 \leqslant i \leqslant n} x_i (f_i + g_i) + (f_0 + g_0) + b^{e_\nu(\overline{x})} \Big(\sum_{1 \leqslant i \leqslant n} x_i (\hat{f}_i + \hat{g}_i) + (\hat{f}_0 + \hat{g}_0) \Big)$$

with $e_\mu(\overline{x})$ abbreviating $\hat{f}(\overline{x}) = 0 \ ? \ g'(\overline{x}) : \big(\hat{g}(\overline{x}) = 0 \ ? \ f'(\overline{x}) : \mathrm{fmin}(f', g')(\overline{x})\big)$ and $e_\nu(\overline{x})$ abbreviating $\hat{f}(\overline{x}) = 0 \ ? \ g'(\overline{x}) : \big(\hat{g}(\overline{x}) = 0 \ ? \ f'(\overline{x}) : \mathrm{fmax}(f', g')(\overline{x})\big)$.

(c) To approximate multiplication of an expression of the form $b^{g'(\overline{x})}$ with $f(\overline{x})$ by an FBI, we may use

$$b^{g'(\overline{x})} \cdot_\mu f(\overline{x}) = \hat{f}(\overline{x}) > 0 \; ? \; \dot{f}(\overline{x}) + b^{f'(\overline{x}) +_\mu g'(\overline{x})}\hat{f}(\overline{x}) : b^{g'(\overline{x})}\dot{f}(\overline{x})$$

$$b^{g'(\overline{x})} \cdot_\nu f(\overline{x}) = \hat{f}(\overline{x}) > 0 \; ? \; b^{f'(\overline{x}) +_\nu g'(\overline{x})}\left(\sum_{1 \leqslant i \leqslant n} x_i(\hat{f}_i + f_i) + (\hat{f}_0 + f_0)\right) : b^{g'(\overline{x})}\dot{f}(\overline{x})$$

(d) Finally we can give approximations for the composition $f(\overline{g})(\overline{x}) = f(g_1(\overline{x}), \dots, g_n(\overline{x}))$:

$$f(\overline{g})_\mu(\overline{x}) = \sum_{1 \leqslant i \leqslant n}^{\mu} g_i(\overline{x}) f_i +_\mu f_0 +_\mu b^{f'(\overline{g})_\mu(\overline{x})} \cdot_\mu \left(\sum_{1 \leqslant i \leqslant n}^{\mu} g_i(\overline{x})\hat{f}_i +_\mu \hat{f}_0\right)$$

The overapproximation $f(\overline{g})_\nu(\overline{x})$ is obtained by replacing μ by ν in the above expression.

(e) Let t be a term and $\mathcal{A}$ an FBI algebra. We define FBIs $\mu_\mathcal{A}(t)$ and $\nu_\mathcal{A}(t)$ such that $\mu_\mathcal{A}(t) = t$ if $t \in \mathcal{V}$ and $\mu_\mathcal{A}(t) = f_\mathcal{A}(\mu_\mathcal{A}(t_1), \dots, \mu_\mathcal{A}(t_n))_\mu$ if $t = f(t_1, \dots, t_n)$. The overapproximation $\nu_\mathcal{A}(t)$ is defined similarly.

Definition 5 yields valid over- and underapproximations.

▶ **Lemma 6.** *If $\mathcal{A}$ is an FBI algebra and t a term then $[\alpha](\mu_\mathcal{A}(t)) \leqslant [\alpha]_\mathcal{A}(t) \leqslant [\alpha](\nu_\mathcal{A}(t))$ for all assignments α.*

The following example illustrates Definition 5.

▶ **Example 7.** We consider the cases for addition and multiplication.

(b) We have $\mathrm{fmin}(x+1, x) = x$ and $\mathrm{fmax}(x+1, x) = x+1$, thus $2^{x+1}y +_\mu 2^x(z+1) = 2^x(y+z+1)$ but $2^{x+1}y +_\nu 2^x(z+1) = 2^{x+1}(y+z+1)$.

In certain pathological cases the approximations of addition are not commutative. To be more precise, the resulting FBIs may be syntactically different but denote the same elementary function. For instance, $2^x \cdot 0 +_\mu 2^{x+1} \cdot 0 = 2^{x+1} \cdot 0$ while $2^{x+1} \cdot 0 +_\mu 2^x \cdot 0 = 2^x \cdot 0$. Still, we do not regard this a problem for our application as the encoding of comparisons takes these cases into account.

(c) For multiplication we have $2^{x+1} \cdot_\mu 2^{2^x} = 2^{(x+1) +_\mu 2^x} = 2^{x+1+2^x}$ and $2^{x+1} \cdot_\nu 2^{2^x} = 2^{x+1+2^x}$, the approximation is thus precise in these cases. On the other hand, as $(x+1) +_\mu 2^x = (x+1) +_\nu 2^x = x+1+2^x$ we have $2^{x+1} \cdot_\mu (z+1+2^{2^x}y) = z+1+2^{x+1+2^x}y$, while $2^{x+1} \cdot_\nu (z+1+2^{2^x}y) = 2^{x+1+2^x}(y+z+1)$.

The following example shows that in practice our approximations are very accurate, i.e., for the motivating example they are exact, i.e., we get the following constraints

$$\begin{array}{ccc} x+5 > x & 2^2 x > 2 & 2^{2^2} > 4 \\ 2x+y+5 > 2x+y+3 & 2^{x+2}y > y+1+2^x 2y & 2^{2^{x+2}} > 2^{x+2+2^x} \\ 2^x(2y+z+1) > 1+2^x(2y+z) & & \end{array}$$

Monotonicity of an FBI $f(\overline{x})$ is expressed by $\mathrm{mon}(f(\overline{x})) = \bigwedge_{1 \leqslant i \leqslant n} \mathrm{mon}_i(f(\overline{x}))$ where $\mathrm{mon}_i(f(\overline{x})) = f_i > 0 \lor \hat{f}_i > 0 \lor (\mathrm{mon}_i(f'(\overline{x})) \land \hat{f}(\overline{x}) > 0)$. An FBI $f(\overline{x})$ is well-defined if $[f(\overline{x}) > 0]$ holds. The main result of this section can now be stated as follows.

▶ **Theorem 8.** *Let $\mathcal{R}$ be a TRS over a signature $\mathcal{F}$ and $\mathcal{A}$ be an FBI algebra on $\mathcal{F}$. If*

$$\bigwedge_{\ell \to r \in \mathcal{R}} [\mu_\mathcal{A}(\ell) > \nu_\mathcal{A}(r)] \land \bigwedge_{f \in \mathcal{F}} \left([f_\mathcal{A}(\overline{x}) > 0] \land \mathrm{mon}(f_\mathcal{A}(\overline{x}))\right)$$

holds then $\mathcal{R}$ is terminating.

method	YES	avg. time	Lescanne's Example	[4, Example 1.1]	[3, Fig. 1]
poly	125	0.3	(0.2)	(0.4)	(0.3)
fbi	41	29.7	*1443.4*	*731.0*	*13540.5*
fbi[d]	170	4.7	16.1	10.0	27.8
fbi[d+]	174	4.2	8.9	7.9	24.1

Table 1 Experimental Results for FBI Algebras.

Finally we address the remaining problem of finding suitable coefficients. To this end we fix the depth of the FBIs used for interpreting function symbols by some heuristics and consider $f_0, \ldots, f_n, \hat{f}_0, \ldots, \hat{f}_n$ in (1) as *unknowns* in the natural numbers. The encodings from before then reduce the search for coefficients to finding models in the SMT logic QF_NIA, i.e., existentially quantified non-linear integer arithmetic, for which tools exist.

3 Limitations

There are TRSs where FBI termination proofs require interpretations of arbitrary depth.

▶ **Example 9.** Let $\mathcal{R}_n$ for $n > 0$ consist of the rules

$$x + \mathsf{0} \to x \qquad x + \mathsf{s}(y) \to \mathsf{s}(x + y) \qquad \mathsf{exp}_0(x) \to x$$
$$\mathsf{exp}_{i+1}(\mathsf{0}) \to \mathsf{exp}_i(\mathsf{s}(\mathsf{0})) \qquad \mathsf{exp}_{i+1}(\mathsf{s}(x)) \to \mathsf{exp}_i(\mathsf{exp}_1(x) + \mathsf{exp}_1(x))$$

for all $0 \leqslant i < n$. Termination of $\mathcal{R}_n$ can be shown by the FBI algebra $\mathcal{A}$ with base $b = 2$ and interpretations $\mathsf{0}_{\mathcal{A}} = 1$, $\mathsf{s}_{\mathcal{A}}(x) = x + 1$, $x +_{\mathcal{A}} y = x + 2y$, and $\mathsf{exp}_{i,\mathcal{A}}(x) = \exp_2^i(2x + 1)$ where $\exp_2^i(x)$ denotes i-fold exponentiation with base 2, i.e., $\exp_2^0(x) = x$ and $\exp_2^{i+1}(x) = 2^{\exp_2^i(x)}$. It is easy to see that any FBI algebra that orients $\mathcal{R}_n$ needs to have at least depth n.

It can be shown that already $\mathcal{R}_1$ admits multiple exponential complexity. As to be expected, actually any TRS compatible with an FBI algebra is bounded by a multiple exponential function. A more precise upper bound is given by the following lemma.

▶ **Lemma 10.** *For any TRS $\mathcal{R}$ compatible with an FBI algebra $\mathcal{A}$ having base b and maximal depth $d - 1$, $\mathrm{dh}_{\mathcal{R}}(n) \in \exp_b^{d\,n}(\mathcal{O}(n))$.*

4 Experimental Results

We implemented FBIs in the termination tool $\mathsf{T_TT_2}$ [2] version 1.15. For experiments[1] we considered the 1463 TRSs in the Standard TRS category of the Termination Problems Data Base (TPDB 8.0.7)[2] and examples from the dedicated literature. If a TRS could not be handled within 60 seconds, the execution of $\mathsf{T_TT_2}$ was aborted.

Table 1 compares the power of FBIs (of depth at most 2) with linear polynomial interpretations when used in direct termination proofs (orient all rules by a single interpretation). For numbers in parentheses $\mathsf{T_TT_2}$ was not successful. The expressions in brackets indicate which heuristics have been used. FBIs as well as linear interpretations use two bits to encode coefficients and seven bits for arithmetic evaluations.

1 Details available from `http://cl-informatik.uibk.ac.at/ttt2/ordinals`
2 Available from `http://termcomp.uibk.ac.at`.

Our experiments show the need for the heuristic limiting the depth of the FBIs (setting [d] in Table 1). We have also experimented with other heuristics [d+] but they are much less effective, i.e., they either slightly decrease the execution time or increase the number of systems shown terminating but are not explained here. The systems where FBIs succeed but linear polynomials fail often require interpretation functions of non-linear shape.

5 Related Work

Lescanne proposed elementary functions for proving (AC-)termination but his implementation is limited to checking the orientation of rules for *given* interpretations [3]. Lucas has achieved partial progress by considering so-called linear elementary interpretations (LEIs) of the shape $A(\overline{x}) + B(\overline{x})^{C(\overline{x})}$ where $A(\overline{x})$, $B(\overline{x})$, and $C(\overline{x})$ are linear polynomials [4]. He proposes an approach based on rewriting, constraint logic programming (CLP), and constraint satisfaction problems (CSPs) to also *find* suitable interpretation functions but leaves an actual implementation of his method as future work.

6 Conclusion

Our findings are related to Problem #28 in the RTA List of Open Problems,[3] which asks to "develop effective methods to decide whether a system decreases with respect to some exponential interpretation". In addition our contribution admits the search for suitable interpretations.

Generalizing elementary interpretations to a non-fixed base is an obvious choice for future work. However, we anticipate that suitable approximations will neither give further deep insights nor significantly improve termination proving power and hence we propose a different line of research, viz., the study how to employ them for AC termination.

Since non-linear polynomials give rise to an exponential size SMT encoding, such interpretations are hardly used within termination tools. We anticipate that suitable approximations could improve the performance of these implementations.

Acknowledgments

We thank the program committee for useful comments.

References

1 F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, Cambridge, 1998.
2 M. Korp, C. Sternagel, H. Zankl, and A. Middeldorp. Tyrolean Termination Tool 2. In *Proc. 20th RTA*, volume 5595 of *LNCS*, pages 295–304, 2009.
3 P. Lescanne. Termination of rewrite systems by elementary interpretations. *Formal Aspects of Computing*, 7(1):77–90, 1995.
4 S. Lucas. Automatic proofs of termination with elementary interpretations. In *Proc. 9th PROLE*, volume 258 of *ENTCS*, pages 41–61, 2009.

[3] `http://www.win.tue.nl/rtaloop/`