Certification of Matrix Interpretations in Coq

Adam Koprowski and Hans Zantema

Eindhoven University of Technology
Department of Computer Science
P.O. Box 513, 5600 MB, Eindhoven, The Netherlands
{A.Koprowski,H.Zantema}@tue.nl

Abstract. We describe how to certify the matrix interpretation method for proving termination of term rewriting in the theorem prover Coq. Certification requires both the formalization of the underlying theory of monotone algebras and their instantiation to vectors and matrices as they are used in the matrix method; as well as a mechanism to deal with concrete examples, based on those theoretical results.

1 Introduction

The *matrix interpretation* method [2] is an instantiation of the well-known approach for proving termination by well-founded orderings, which turned out to be powerful in practice. *CoLoR* [1] is a Coq library on rewriting and termination made with the goal of certification of termination proofs in mind. This goal is accomplished by *Rainbow* which consists of the description of termination proofs in XML format and a mechanism to transform proofs in this format into Coq scripts that certify their correctness (using theoretical results from CoLoR).

We succeeded in certifying the matrix interpretation method in Coq. We did that by formalizing the theory of monotone algebras and then instantiating it to the algebra of vectors used in the matrix interpretation method. Then we constructed a framework within Coq to check that all the conditions of the respective theorems are fulfilled in the instantiation to concrete examples. Finally we extended Rainbow with a proof format for the matrix interpretation method and with the ability to translate proofs in this format to certified Coq proofs.

2 Monotone algebras

Here we summarize the monotone algebra theory as presented in [2]. There is one difference: in contrast to [2] we do not consider many-sortedness. It is not essential for certification as every proof in the many-sorted setting can be trivially translated to one-sorted setting. The reason for this more complex setup in [2] is that it allows for an optimization in the search for termination proofs using matrix interpretations.

The monotone algebra approach works for all non-empty sets A; when using the matrix method A always consists of the set of vectors over $\mathbf{N}$.

Definition 1. *An operation* $[f] : A \times \cdots \times A \to A$ *is* monotone *with respect to a binary relation* $\to$ *on* A *if for all* $a_i, b_i \in A$ *for* $i = 1, \ldots, n$ *with* $a_i \to b_i$ *for some* i *and* $a_j = b_j$ *for all* $j \neq i$ *we have* $[f](a_1, \ldots, a_n) \to [f](b_1, \ldots, b_n)$.

A weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ *is a* Σ*-algebra* $(A, [\cdot])$ *equipped with two binary relations* $>$, $\gtrsim$ *on* A *such that*

- $>$ *is well-founded;*
- $> \cdot \gtrsim \subseteq >$*;*
- *for every* $f \in \Sigma$ *the operation* $[f]$ *is monotone with respect to* $\gtrsim$.

An extended monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ *is a weakly monotone* Σ*-algebra* $(A, [\cdot], >, \gtrsim)$ *in which moreover for every* $f \in \Sigma$ *the operation* $[f]$ *is monotone with respect to* $>$.

We write $\mathsf{SN}(\to_{\mathcal{R}}/\to_{\mathcal{S}})$ for termination of $\mathcal{R}$ relative to $\mathcal{S}$, meaning that every infinite $\mathcal{R} \cup \mathcal{S}$ reduction consists of only finitely many $\mathcal{R}$-steps. We write $\mathsf{SN}(\to_{\mathcal{R}_{\mathrm{top}}}/\to_{\mathcal{S}})$ if $\mathcal{R}$-steps are only allowed at the root position. Such relative top termination plays a crucial role in the dependency pair setting as $\mathsf{SN}(\to_{\mathcal{R}})$ is equivalent to $\mathsf{SN}(\to_{\mathsf{DP}(\mathcal{R})_{\mathrm{top}}}/\to_{\mathcal{R}})$. Up to presentation details the following theorem is the one-sorted version of the main theorem for the matrix interpretations from [2, Theorem 2].

Theorem 2. *Let* R, R', S, S' *be TRSs over a signature* Σ.

1. *Let* $(A, [\cdot], >, \gtrsim)$ *be an extended monotone* Σ*-algebra such that* $[\ell, \alpha] \gtrsim [r, \alpha]$ *for every rule* $\ell \to r$ *in* $\mathcal{R} \cup \mathcal{S}$ *and* $[\ell, \alpha] > [r, \alpha]$ *for every rule* $\ell \to r$ *in* $R' \cup S'$*, for every* $\alpha : \mathcal{X} \to A$.
   Then $\mathsf{SN}(\to_{\mathcal{R}}/\to_{\mathcal{S}})$ *implies* $\mathsf{SN}(\to_{\mathcal{R}} \cup \to_{\mathcal{R}'} / \to_{\mathcal{S}} \cup \to_{\mathcal{S}'})$.
2. *Let* $(A, [\cdot], >, \gtrsim)$ *be a weakly monotone* Σ*-algebra such that* $[\ell, \alpha] \gtrsim [r, \alpha]$ *for every rule* $\ell \to r$ *in* $\mathcal{R} \cup \mathcal{S}$ *and* $[\ell, \alpha] > [r, \alpha]$ *for every rule* $\ell \to r$ *in* $\mathcal{R}'$*, for every* $\alpha : \mathcal{X} \to A$.
   Then $\mathsf{SN}(\to_{\mathcal{R}_{\mathrm{top}}}/\to_{\mathcal{S}})$ *implies* $\mathsf{SN}((\to_{\mathcal{R}} \cup \to_{\mathcal{R}'})_{\mathrm{top}}/\to_{\mathcal{S}})$.

We obviously have $\mathsf{SN}(\to_{\mathcal{R}}/\to_{\mathcal{S}})$ if and only if the relation $\to_{\mathcal{S}}^* \cdot \to_{\mathcal{R}}$ is well-founded. Our formalized proof of Theorem 2 is constructive (hence slightly differs from the proof in [2]) and is based on the following lemma.

Lemma 3. *Let* $\to_{\mathcal{R}}, \to_{\mathcal{S}}, \to_{\mathcal{R}'}, \to_{\mathcal{S}'}$ *be binary relations for which* $\to_{\mathcal{S}}^* \cdot \to_{\mathcal{R}}$ *and* $(\to_{\mathcal{R}} \cup \to_{\mathcal{S}})^* \cdot (\to_{\mathcal{R}'} \cup \to_{\mathcal{S}'})$ *are well-founded. Then* $(\to_{\mathcal{S}} \cup \to_{\mathcal{S}'})^* \cdot (\to_{\mathcal{R}} \cup \to_{\mathcal{R}'})$ *is well-founded.*

When thinking in terms of infinite sequences this lemma can easily be proven by truncating the initial part of such, supposedly, infinite sequences and observing that the remaining part must be finite. Working in the constructive setting of Coq forces us to use a slightly different kind of reasoning where the focus is on providing a relation that is decreasing along the sequence and performing induction with respect to it. Our experience shows, however, that for typical properties on well-foundedness, including this lemma, the ingredients of reasoning with infinite sequences can be recognized in such constructive proofs.

3 Matrix interpretations

In this section we show how monotone algebras are instantiated for the matrix interpretations with a fixed dimension d.

For the interpretation $[f]$ of a symbol $f \in \Sigma$ of arity n we choose n matrices $F_1, F_2, \ldots, F_n$ over $\mathbf{N}$, each of size $d \times d$, such that the upper left elements $(F_i)_{1,1}$ are positive for all $i = 1, 2, \ldots, n$, and a vector $\boldsymbol{f} \in \mathbf{N}^d$. Now we define $[f](\boldsymbol{v_1}, \ldots, \boldsymbol{v_n}) = F_1\boldsymbol{v_1} + \cdots + F_n\boldsymbol{v_n} + \boldsymbol{f}$ for all $\boldsymbol{v_1}, \ldots, \boldsymbol{v_n} \in A$.

So we fix a monotone algebra with $A = \mathbf{N}^d$, interpretations $[\cdot]$ defined as above and we use the following orders on algebra elements:

$$(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \iff \forall i : u_i \geq_{\mathbf{N}} v_i$$
$$(u_1, \ldots, u_d) > (v_1, \ldots, v_d) \iff (u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbf{N}} v_1$$

Let $x_1, \ldots, x_k$ be the variables occurring in ℓ, r. Then due to the linear shape of the functions $[f]$ we can compute matrices $L_1, \ldots, L_k, R_1, \ldots, R_k$ and vectors $\boldsymbol{l}, \boldsymbol{r}$ such that

$$[\ell, \alpha] = L_1\boldsymbol{x_1} + \cdots + L_k\boldsymbol{x_k} + \boldsymbol{l}$$
$$[r, \alpha] = R_1\boldsymbol{x_1} + \cdots + R_k\boldsymbol{x_k} + \boldsymbol{r}$$

where $\alpha(x_i) = \boldsymbol{x_i}$ for $i = 1, \ldots, k$.

For matrices $B, C \in \mathbf{N}^{d \times d}$ write

$$B \succcurlyeq C \iff \forall i, j : (B)_{i,j} \geq (C)_{i,j}.$$

The following lemma provides a decision procedure for orders $>$ and $\gtrsim$ lifted to terms as required by the conditions of Theorem 2.

Lemma 4. *Let ℓ, r be terms and let matrices $L_1, \ldots, L_k, R_1, \ldots, R_k$ and vectors $\boldsymbol{l}, \boldsymbol{r}$ be defined as above. Then:*

- $\forall \alpha : \mathcal{X} \to A, [\ell, \alpha] \gtrsim [r, \alpha] \iff \boldsymbol{l} \gtrsim \boldsymbol{r} \wedge \forall i : L_i \succcurlyeq R_i$, *and*
- $\forall \alpha : \mathcal{X} \to A, [\ell, \alpha] > [r, \alpha] \iff \boldsymbol{l} > \boldsymbol{r} \wedge \forall i : L_i \succcurlyeq R_i$.

4 Coq formalization

In this section we briefly outline the Coq formalization. It consists of four parts that we will discuss in turn: the library on matrices, the theory of monotone algebras, the theory of the matrix interpretations and the extension of Rainbow to handle proofs using the matrix interpretation technique.

We developed a simple library on matrices as a functor, which takes a semiring of coefficients as an argument and produces a structure of matrices of arbitrary dimension with such coefficients along with basic operations (matrix construction, addition, multiplication, transposition etc.) and some basic properties (commutativity of addition, associativity of multiplication etc.). Matrices

are represented as vectors of vectors which enables us to reuse the rich library on vectors that is part of CoLoR.

The theory of monotone algebras is developed as another functor that takes a carrier set A, a signature Σ with interpretations for all $f \in \Sigma$, two relations $>$ and $\gtrsim$ (which are typically orders but this is not required) and all properties as presented in Section 2. To be able to deal with concrete examples we need one additional ingredient: a decision procedure for liftings of $>$ and $\gtrsim$ to terms, as we must be able to check whether a given rule is (weakly) oriented. In fact instead of requiring a decision procedure for such relations we require it for some subsets of them, which are then used to prove termination. This is because in some instances the relations in question are undecidable (for example for non-linear polynomial interpretations) and in this way we are able to use some heuristics in such cases. For the matrix interpretations the intended relations are decidable, due to Lemma 4, but this allows us not to prove completeness of this characteristic (so we only had to prove the 'if' parts of this lemma). The module generated by this functor contains all the results (including Theorem 2) and Coq tactics needed to prove termination of concrete examples.

Finally the matrix interpretations are built as yet another functor that expects as argument a structure containing: a signature Σ, a dimension d for vectors and matrices and the matrix interpretations for all $f \in \Sigma$. It instantiates monotone algebras to matrix interpretations of dimension d. The main difficulty is providing a decision procedure for the intended relations, which accounts for proving the 'if' part of Lemma 4.

Finally we extended the Rainbow XML proof format with a format for the matrix interpretations proofs and with the ability to translate such proofs to actual Coq proofs. We added support for this format to the termination prover TPA [3] and with its help obtained certified termination proofs for 237 TRSs from TPDB v. 3.2. By expressing the CoLoR implementation of polynomial interpretations, by Sébastien Hinderer, in the setting of monotone algebras and by evaluating (on the same problem set) matrix interpretations combined with (non-linear) polynomial interpretations, 275 proofs were obtained. The average time for verification of the correctness of a proof by Coq (excluding proof search by TPA) was 5 seconds.

References

1. F. Blanqui, S. Coupet-Grimal, W. Delobel, S. Hinderer and A. Koprowski, CoLoR, a Coq Library on Rewriting and termination. In *WST '06*.
2. J. Endrullis, J. Waldmann, and H. Zantema. Matrix interpretations for proving termination of term rewriting. In *IJCAR '06*, volume 4130 of *LNCS*, pages 574–588.
3. A. Koprowski TPA: Termination Proved Automatically. In *RTA '06*, volume 4098 of *LNCS*, pages 257–266.