Science of Computer Programming 111 (2015) 110–134

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

A unified ordering for termination proving

Akihisa Yamada [a,b,*], Keiichirou Kusakari [c], Toshiki Sakabe [a]

[a] Graduate School of Information Science, Nagoya University, Japan
[b] National Institute of Advanced Industrial Science and Technology, Japan
[c] Faculty of Engineering, Gifu University, Japan

A R T I C L E   I N F O

Article history:
Received 11 December 2013
Received in revised form 7 July 2014
Accepted 9 July 2014
Available online 14 August 2014

Keywords:
Term rewriting
Reduction order
Termination

A B S T R A C T

We introduce a reduction order called the weighted path order (WPO) that subsumes many existing reduction orders. WPO compares weights of terms as in the Knuth–Bendix order (KBO), while WPO allows weights to be computed by a wide class of interpretations. We investigate summations, polynomials and maximums for such interpretations. We show that KBO is a restricted case of WPO induced by summations, the polynomial order (POLO) is subsumed by WPO induced by polynomials, and the lexicographic path order (LPO) is a restricted case of WPO induced by maximums. By combining these interpretations, we obtain an instance of WPO that unifies KBO, LPO and POLO. In order to fit WPO in the modern dependency pair framework, we further provide a reduction pair based on WPO and partial statuses. As a reduction pair, WPO also subsumes matrix interpretations. We finally present SMT encodings of our techniques, and demonstrate the significance of our work through experiments.

1. Introduction

Proving *termination* of *term rewrite systems (TRSs)* is one of the most important tasks in program verification and automated theorem proving, where *reduction orders* play a fundamental role. The classic use of reduction orders in termination proving is illustrated in the following example:

Example 1. Consider the following TRS $\mathcal{R}_{\mathsf{fact}}$:

$$\mathcal{R}_{\mathsf{fact}} := \begin{cases} \mathsf{fact}(0) \rightarrow \mathsf{s}(0) \\ \mathsf{fact}(\mathsf{s}(x)) \rightarrow \mathsf{s}(x) * \mathsf{fact}(x) \end{cases}$$

which defines the factorial function, provided the binary symbol $*$ is defined as multiplication. We can prove termination of $\mathcal{R}_{\mathsf{fact}}$ by finding a reduction order $\succ$ that satisfies the following constraints:

$$\mathsf{fact}(0) \succ \mathsf{s}(0)$$

$$\mathsf{fact}(\mathsf{s}(x)) \succ \mathsf{s}(x) * \mathsf{fact}(x)$$

* Correspondence to: Graduate School of Information Science, Nagoya University, Japan.
E-mail addresses: akihisa.yamada@aist.go.jp (A. Yamada), kusakari@gifu-u.ac.jp (K. Kusakari), sakabe@is.nagoya-u.ac.jp (T. Sakabe).

http://dx.doi.org/10.1016/j.scico.2014.07.009

A number of reduction orders have been proposed, and their efficient implementation is demonstrated by several automatic termination provers such as AProVE [1] or $\mathsf{T_TT_2}$ [2].

One of the most well-known reduction orders is the *lexicographic path order (LPO)* of Kamin and Lévy [3], a variant of the *recursive path order (RPO)* of Dershowitz [4]. LPO is unified with RPO using *status* [5]. Recently, Codish et al. [6] proposed an efficient implementation using a SAT solver for termination proving by RPO with status.

The *Knuth–Bendix order (KBO)* [7] is the oldest reduction order. KBO has become a practical alternative in automatic termination checking since Korovin and Voronkov [8] discovered a polynomial-time algorithm for termination proofs with KBO. Zankl et al. [9] proposed another implementation method via SAT/SMT encoding, and verified a significant improvement in efficiency over dedicated implementations of the polynomial-time algorithm. However, KBO is disadvantageous compared to LPO when *duplicating* rules (where a variable occurs more often in the right-hand side than in the left-hand side) are considered. Actually, no duplicating rule can be oriented by KBO. To overcome this disadvantage, Middeldorp and Zantema [10] proposed the *generalized KBO (GKBO)*, which generalizes weights over algebras that are weakly monotone and *strictly simple*: $f(\dots, x, \dots) > x$. Ludwig and Waldmann proposed another extension of KBO called the *transfinite KBO (TKBO)* [11–13], which extends the weight function to allow linear polynomials over ordinals. However, proving termination with TKBO involves solving the satisfiability problem of non-linear arithmetic which is undecidable in general. Moreover, TKBO still does not subsume LPO.

The *polynomial order (POLO)* of Lankford [14] interprets each function symbol by a strictly monotone polynomial. Zantema [15] extended the method to algebras and suggested combining the "max" operator with polynomial interpretations (*max-polynomials* in terms of [16]). Fuhs et al. proposed an efficient SAT encoding of POLO in [17], and a general version of POLO with max in [16].

The *dependency pair (DP) method* of Arts and Giesl [18] significantly enhances the classic approach of reduction orders by analyzing cyclic dependencies between rewrite rules. In the DP method, reduction orders are extended to *reduction pairs* $\langle \succsim, \succ \rangle$, and it suffices if one rule in a recursive dependency is strictly oriented, and other rules are only weakly oriented.[1]

Example 2. Consider again the TRS $\mathcal{R}_{\mathsf{fact}}$. There is one cyclic dependency in $\mathcal{R}_{\mathsf{fact}}$, that is represented by the *dependency pair* $\mathsf{fact}^\sharp(\mathsf{s}(x)) \to \mathsf{fact}^\sharp(x)$, where $\mathsf{fact}^\sharp$ is a fresh symbol. We can prove termination of $\mathcal{R}_{\mathsf{fact}}$ by finding a reduction pair $\langle \succsim, \succ \rangle$ that satisfies the following constraints[2]:

$$\begin{aligned} \mathsf{fact}^\sharp(\mathsf{s}(x)) &\succ \mathsf{fact}^\sharp(x) \\ \mathsf{fact}(\mathsf{0}) &\succsim \mathsf{s}(\mathsf{0}) \\ \mathsf{fact}(\mathsf{s}(x)) &\succsim \mathsf{s}(x) * \mathsf{fact}(x) \end{aligned}$$

One of the typical methods for designing reduction pairs is *argument filtering* [18], which generates reduction pairs from arbitrary reduction orders. Hence, reduction orders are still an important subject to study in modern termination proving. Another typical technique is generalizing interpretation methods to *weakly* monotone ones, e.g. allowing 0 coefficients for polynomial interpretations [18]. Endrullis et al. [21] extended polynomial interpretations to *matrix interpretations*, and presented their implementation via SAT encoding. More recently, Bofill et al. [22] proposed a reduction pair called *RPOLO*, which unifies standard POLO and RPO by choosing either *RPO-like* or *POLO-like* comparison depending on function symbols.

These reduction orders and reduction pairs require different correctness proofs and different implementations. In this paper, we extract the underlying essence of these reduction orders and introduce a general reduction order called the *weighted path order (WPO)*. Technically, WPO is a further generalization of GKBO that relaxes the strict simplicity condition of weights to *weak simplicity*. This relaxation becomes possible by combining the recursive checks of LPO with GKBO. While strict simplicity is so restrictive that GKBO does not even subsume the standard KBO, weak simplicity is so general that WPO subsumes not only KBO but also most of the reduction orders described above (LPO, TKBO, POLO and so on), except for matrix interpretations which are not weakly simple in general.

There exist several earlier works on generalizing existing reduction orders. The *semantic path order (SPO)* of Kamin and Lévy [3] is a generalization of RPO where precedence comparison is generalized to an arbitrary well-founded order on terms. However, to prove termination by SPO users have to ensure monotonicity by themselves, even if the underlying well-founded order is monotone (cf. [23]). On the other hand, monotonicity of WPO is guaranteed. Borralleras et al. [23] proposed the *monotonic SPO (MSPO)* that ensures monotonicity by using an external monotonic order. As far as we know, no external order proposed so far can generate WPO as MSPO. Moreover, WPO can be used as such an external order. The *general path order (GPO)* [24,25] is a very general framework by which many reduction orders are subsumed. Due to the generality, however, implementing GPO seems to be quite challenging. Indeed, we are not aware of any tool that implements GPO.

---

1 The technique is called the *dependency graph refinement* in [18], and inherited in the *DP framework* [19,20], a successor of the DP method.

2 The last two constraints can be removed by considering *usable rules* [18].