

DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN
UNIVERSIDAD POLITÉCNICA DE VALENCIA

P.O. Box: 22012 E-46021 Valencia (SPAIN)



Informe Técnico / Technical Report

Ref. No.: DSIC-II/04/08	Pages: 22
Title: Modular Termination of Basic Narrowing	
Author(s): María Alpuente and Santiago Escobar and José Iborra	
Date: April 2008	
Keywords: Narrowing, Rewriting, Termination, Modularity	

Vº Bº
Leader of research Group

Author(s)

Modular Termination of Basic Narrowing^{*}

María Alpuente, Santiago Escobar, and José Iborra

Universidad Politécnica de Valencia, Spain.
{alpuente,sescobar,jiborra}@dsic.upv.es

Abstract. Basic narrowing is a restricted form of narrowing which constrains narrowing steps to a set of non-blocked (or basic) positions. Basic narrowing has a number of important applications including equational unification in canonical theories. Another application is analyzing termination of narrowing by checking the termination of basic narrowing, as done in pioneering work by Hullot. In this work, we study the modularity of termination of basic narrowing in hierarchical combinations of TRSs, including a generalization of proper extensions with shared subsystem. This provides new algorithmic criteria to prove termination of basic narrowing.

1 Introduction

Narrowing [12] is a generalization of term rewriting that allows free variables in terms (as in logic programming) and replaces pattern matching with syntactic unification. Narrowing was originally introduced as a mechanism for solving equational unification problems [16], hence termination results for narrowing have been traditionally achieved as a by-product of addressing the decidability of equational unification. Basic narrowing [16] is a refinement of narrowing which restricts narrowing steps to a set of non-blocked (or basic) positions, and is still complete for equational unification in canonical TRSs. Termination of basic narrowing was first studied by Hullot in [16], where a faulty termination result for narrowing was enunciated, namely the termination of all narrowing derivations in canonical theories when all basic narrowing derivations issuing from the right-hand sides (rhs's) of the rules terminate. This result was implicitly corrected in [17], downgrading it to the more limited result of basic narrowing termination (instead of ordinary narrowing) under the basic narrowing termination requirement for the rhs's of the rules. The missing condition to recover narrowing termination in [16] is to require that the TRS satisfies Réty's maximal commutation condition for narrowing sequences [24], as we proved¹ in [2]. From this result, we also distilled in [2] a syntactic characterization of TRSs where

^{*} This work has been partially supported by the EU (FEDER) and Spanish MEC project TIN2007-68093-C02-02, Integrated Action Hispano-Alemana A2006-0007, and UPV-VIDI grant 3249 PAID0607.

¹ We also explicitly dropped in [2] the superfluous requirement of canonicity from Hullot's termination result, as cognoscenti tacitly do.

termination of basic narrowing implies termination of narrowing, namely right-linear TRSs that are either left-linear or regular and where narrowing computes² only normalized substitutions.

The main motivation for this paper is proving termination of narrowing via termination of basic narrowing. We present several criteria for modular termination of basic narrowing in hierarchical combinations of TRSs, including generalized proper extensions with shared subsystem. By adopting the divide-and-conquer principle, this allows us to prove (basic) narrowing termination in a modular way, thus extending the class of TRSs for which termination of basic narrowing (and hence termination of narrowing) can be proved. We assume a standard notion of modularity, where a property φ of TRSs is called *modular* if, whenever \mathcal{R}_1 and \mathcal{R}_2 satisfy φ , then their combination $\mathcal{R}_1 \cup \mathcal{R}_2$ also satisfies φ . Our modularity results for basic narrowing rely on a commutation result for basic narrowing sequences that has not been identified in the related literature before.

In [22], a modularity result for decidability of unification (via termination of narrowing) in canonical TRSs is given. However, this result does not imply the modularity of narrowing termination for a particular class of TRSs but rather the possibility to define a terminating, modular narrowing procedure. Namely, the result in [22] is as follows: given a canonical TRS \mathcal{R} such that narrowing terminates for \mathcal{R}_1 and \mathcal{R}_2 and $\mathcal{R} \downarrow \subseteq \mathcal{R}_1 \downarrow \mathcal{R}_2 \downarrow$ (i.e. normalization with $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$ can be obtained by first normalizing with \mathcal{R}_1 followed by a normalization with \mathcal{R}_2), then there is a terminating and complete, modular narrowing strategy for \mathcal{R} . Any complete strategy can be used within the modular procedure given in [22], including the basic narrowing strategy. As far as we know this is the only previous modularity result in the literature that concerns the modular termination of basic narrowing.

After some preliminaries in Section 2, we study commutation properties of basic narrowing derivations in Section 3. Section 4 recalls some standard notions for modularity of rewriting and presents our main modularity results for the termination of basic narrowing. In order to prove in Section 4.5 that termination of basic narrowing is modular for proper extensions [23], we first prove an intermediate result: in Section 4.4 we prove that basic narrowing termination is modular for a restriction of proper extensions called *nice* extensions [23]. In Section 5 we generalize our results and prove modularity for a wider class of TRSs called *relaxed proper extensions*. We conclude in Section 6. Proofs of all results in this paper are included in [1].

2 Preliminaries

In this section, we briefly recall the essential notions and terminology of term rewriting [9, 21, 25].

\mathcal{V} denotes a countably infinite set of variables, and Σ denotes a set of function symbols, or signature, each of which has a fixed associated arity. Terms are

² This includes some popular classes of TRSs, including linear constructor systems

viewed as labelled trees in the usual way, where $\mathcal{T}(\Sigma, \mathcal{V})$ and $\mathcal{T}(\Sigma)$ denote the non-ground term algebra and the ground algebra built on $\Sigma \cup \mathcal{V}$ and Σ , respectively. Positions are defined as sequences of positive natural numbers used to address subterms of a term, with ϵ as the root (or top) position (i.e., the empty sequence). Concatenation of positions p and q is denoted by $p.q$, and $p < q$ is the usual prefix ordering. Given $S \subseteq \Sigma \cup \mathcal{V}$, $\text{Pos}_S(t)$ denotes the set of positions of a term t that are rooted by function symbols or variables in S . $\text{Pos}_{\{f\}}(t)$ with $f \in \Sigma \cup \mathcal{V}$ will be simply denoted by $\text{Pos}_f(t)$, and $\text{Pos}_{\Sigma \cup \mathcal{V}}(t)$ will be simply denoted by $\text{Pos}(t)$. $t|_p$ is the subterm at the position p of t . $t[s]_p$ is the term t with the subterm at the position p replaced with term s . By $\text{Var}(s)$, we denote the set of variables occurring in the syntactic object s . By \bar{x} , we denote a tuple of pairwise distinct variables. A *fresh* variable is a variable that appears nowhere else. A *linear* term is one where every variable occurs only once.

A *substitution* σ is a mapping from the set of variables \mathcal{V} into the set of terms $\mathcal{T}(\Sigma, \mathcal{V})$, with a finite domain $D(\sigma)$ and image $I(\sigma)$. A substitution is represented as $\{x_1/t_1, \dots, x_n/t_n\}$ for variables x_1, \dots, x_n and terms t_1, \dots, t_n . The application of substitution θ to term t is denoted by $t\theta$, using postfix notation. Composition of substitutions is denoted by juxtaposition, i.e., the substitution $\sigma\theta$ denotes $(\theta \circ \sigma)$. We write $\theta|_{\text{Var}(s)}$ to denote the restriction of the substitution θ to the set of variables in s ; by abuse of notation, we often simply write $\theta|_s$. Given a term t , $\theta = \nu[t]$ iff $\theta|_{\text{Var}(t)} = \nu|_{\text{Var}(t)}$, that is, $\forall x \in \text{Var}(t), x\theta = x\nu$. A substitution θ is more general than σ , denoted by $\theta \leq \sigma$, if there is a substitution γ such that $\theta\gamma = \sigma$. A *unifier* of terms s and t is a substitution ϑ such that $s\vartheta = t\vartheta$. The *most general unifier* of terms s and t , denoted by $\text{mgu}(s, t)$, is a unifier θ such that for any other unifier θ' , $\theta \leq \theta'$.

A *term rewriting system* (TRS) \mathcal{R} is a pair (Σ, R) , where R is a finite set of rewrite rules of the form $l \rightarrow r$ such that $l, r \in \mathcal{T}(\Sigma, \mathcal{V})$, $l \notin \mathcal{V}$, and $\text{Var}(r) \subseteq \text{Var}(l)$. We will often write just \mathcal{R} or (Σ, R) instead of $\mathcal{R} = (\Sigma, R)$. Given a TRS $\mathcal{R} = (\Sigma, R)$, the signature Σ is often partitioned into two disjoint sets $\Sigma = \mathcal{C} \uplus \mathcal{D}$, where $\mathcal{D} = \{f \mid f(t_1, \dots, t_n) \rightarrow r \in R\}$ and $\mathcal{C} = \Sigma \setminus \mathcal{D}$. Symbols in \mathcal{C} are called *constructors*, and symbols in \mathcal{D} are called *defined functions*. The elements of $\mathcal{T}(\mathcal{C}, \mathcal{V})$ are called *constructor terms*. We let $\text{Def}(\mathcal{R})$ denote the set of defined symbols in \mathcal{R} . A rewrite step is the application of a rewrite rule to an expression. A term $s \in \mathcal{T}(\Sigma, \mathcal{V})$ *rewrites* to a term $t \in \mathcal{T}(\Sigma, \mathcal{V})$, denoted by $s \xrightarrow{p}_{\mathcal{R}} t$, if there exist $p \in \text{Pos}_{\Sigma}(s)$, $l \rightarrow r \in \mathcal{R}$, and substitution σ such that $s|_p = l\sigma$ and $t = s[r\sigma]_p$. When no confusion can arise, we omit the subscript in $\rightarrow_{\mathcal{R}}$. We also omit the reduced position p when it is not relevant. A term s is a *normal form* w.r.t. the relation $\rightarrow_{\mathcal{R}}$ (or simply a normal form), if there is no term t such that $s \rightarrow_{\mathcal{R}} t$. A term is a reducible expression or *redex* if it is an instance of the left hand side of a rule in \mathcal{R} . A term s is a *head normal form* if there are no terms t, t' s.t. $s \rightarrow_{\mathcal{R}}^* t' \xrightarrow{\epsilon}_{\mathcal{R}} t$. A TRS \mathcal{R} is (\rightarrow) -*terminating* (also called strongly normalizing or noetherian) if there are no infinite reduction sequences $t_1 \rightarrow_{\mathcal{R}} t_2 \rightarrow_{\mathcal{R}} \dots$.

Narrowing is a symbolic computation mechanism that generalizes rewriting by replacing pattern matching with syntactic unification. Many redundancies in

the narrowing algorithm can be eliminated by restricting narrowing steps to a distinguished set of *basic* positions, which was proposed by Hullot in [16].

2.1 Basic Narrowing

Basic narrowing is the restriction of narrowing introduced by Hullot [16] which is essentially based on forbidding narrowing steps on terms brought in by instantiation. We use the definition of basic narrowing given in [15], where the expression to be narrowed is split into a *skeleton* t and an *environment* part θ , i.e., $\langle t, \theta \rangle$. The environment part keeps track of the accumulated substitution so that, at each step, substitutions are composed in the environment part, but are not applied to the expression in the skeleton part, as opposed to ordinary narrowing. For TRS \mathcal{R} , $l \rightarrow r \ll \mathcal{R}$ denotes that $l \rightarrow r$ is a *fresh* variant of a rule in \mathcal{R} , i.e., all the variables are *fresh*.

Definition 1 (Basic narrowing). [15] *Given a term $s \in \mathcal{T}(\Sigma, \mathcal{V})$ and a substitution σ , a basic narrowing step for $\langle s, \sigma \rangle$ is defined by $\langle s, \sigma \rangle \xrightarrow{b}_{p, \mathcal{R}, \theta} \langle t, \sigma' \rangle$ if there exist $p \in \text{Pos}_\Sigma(s)$, $l \rightarrow r \ll \mathcal{R}$, and substitution θ such that $\theta = \text{mgu}(s|_p \sigma, l)$, $t = (s[r]_p)$, and $\sigma' = \sigma\theta$.*

Along a basic narrowing derivation, the set of *basic* occurrences of $\langle t, \theta \rangle$ is $\text{Pos}_\Sigma(t)$, and the non-basic occurrences are $\text{Pos}_\Sigma(t\theta) - \text{Pos}_\Sigma(t)$. When p is not relevant, we simply denote the basic narrowing relation by $\xrightarrow{b}_{\mathcal{R}, \theta}$. By abuse of notation, also we often relax the skeleton-environment notation for basic narrowing steps, i.e., $\langle s, \sigma \rangle \xrightarrow{b}_{\mathcal{R}, \theta} \langle t, \sigma' \rangle$ and use the more compact notation $s\sigma \xrightarrow{b}_{\mathcal{R}, \theta} t\sigma\theta$ instead, but then suitable track of the basic positions along the narrowing sequences is implicitly done.

We say that \mathcal{R} is (\xrightarrow{b}) -terminating when every basic narrowing derivation issuing from any term terminates. All modular termination results in this paper are based on the following termination result for basic narrowing. It is essentially Hullot's basic narrowing termination result, where we have explicitly dropped the superfluous requirement of canonicity [2].

Theorem 1 (Termination of Basic Narrowing). [16, 2] *Let \mathcal{R} be a TRS. If for every $l \rightarrow r \in \mathcal{R}$, all basic narrowing derivations issuing from r terminate, then \mathcal{R} is (\xrightarrow{b}) -terminating.*

In the literature, this condition has been approximated by requiring that every rhs of a rewrite rule is a constructor term [22]. This approximation has been generalized in [2] by requiring the rhs's to be a *rigid normal form* (rnf), i.e., unnarrowable.

3 A Commutation result for basic narrowing derivations

The commutation properties of ordinary narrowing were extensively studied by Rety in [24]. We analyze here those of basic narrowing, in Rety's style. First let us recall the notion of *antecedent* of a position in a rewriting sequence [24].

Definition 2 (Antecedent of a position). [24] Let $t \xrightarrow{p}_{l \rightarrow r} t'$ be a rewriting step, $v \in \text{Pos}(t)$, and $v' \in \text{Pos}(t')$. We say position v is an antecedent of v' iff either

1. $v \parallel p$, i.e., v and p are disjoint, and $v = v'$, or
2. there exists an occurrence $u' \in \text{Pos}_x(r)$ of a variable x in r s.t. $v' = p.u'.w$ and $v = p.u.w$, where $u \in \text{Pos}_x(l)$ is an occurrence of x in l .

With the notations of the previous definition, we have:

1. $t|_v = t'|_{v'}$,
2. v' may have no antecedent if $v' = p.u'$ with $u' \in \text{Pos}_\Sigma(r)$, or if $v' < p$,

This notion extends to a rewrite sequence by transitive closure of the rewriting relation in the usual way. The notion of antecedent can also be extended to narrowing sequences as follows.

Definition 3 (Narrowing antecedent of a position). [24] Let $t \xrightarrow{b}_{\mathcal{R}, \sigma}^* t'$, $v \in \text{Pos}(t)$, and $v' \in \text{Pos}(t')$. We say v is an antecedent of v' iff v is an antecedent of v' in the rewrite sequence $t\sigma \rightarrow_{\mathcal{R}}^* t'$.

And note that now we have $t|_v\sigma = t'|_{v'}$. In the following, we consider basic narrowing derivations of the form

$$s \xrightarrow{b}_{p, g \rightarrow d, \sigma} t \xrightarrow{b}_{q, l \rightarrow r, \theta} u \quad (1)$$

and we are interested in the conditions that allow us to commute the first two steps by first applying to s the rule $l \rightarrow r$ and then the rule $g \rightarrow d$ to the resulting term. If the subterm $t|_q$ already exists in s , i.e., if q admits at least one antecedent in s , the idea essentially consists in applying $l \rightarrow r$ to all the antecedents of q , and then applying $g \rightarrow d$ to the resulting term. Let us give an example for motivation.

Example 1. [24] Let us consider the following TRS \mathcal{R}_4 :

$$\mathcal{R}_4 = \begin{cases} \mathbf{f}(x, x) \rightarrow x & (\text{r1}) \\ \mathbf{g}(x, \mathbf{h}(x)) \rightarrow x & (\text{r2}) \end{cases}$$

and the basic narrowing derivation:

$$\langle \mathbf{h}(\mathbf{f}(0, x), \mathbf{g}(x, y)), \{\} \rangle \xrightarrow{b}_{p=1, r1, \{x/0\}} \langle \mathbf{h}(x, \mathbf{g}(x, y)), \{x/0\} \rangle \xrightarrow{b}_{q=2, r2, \{y/\mathbf{h}(x)\}} \langle \mathbf{h}(x, x), \{x/0\} \rangle.$$

The occurrences p and q are disjoint, therefore q has an antecedent in s at $p_0 = 2$. By first applying $r2$ at $p_0 = 2$, and then $r1$ at p we get:

$$\langle \mathbf{h}(\mathbf{f}(0, x), \mathbf{g}(x, y)), \{\} \rangle \xrightarrow{b}_{p_0=2, r2, \{y/\mathbf{h}(x)\}} \langle \mathbf{h}(\mathbf{f}(0, x), x), \{\} \rangle \xrightarrow{b}_{p=1, r1, \{x/0\}} \langle \mathbf{h}(x, x), \{x/0\} \rangle$$

The following result establishes that in a basic narrowing derivation, the antecedent of a position is always in the skeleton part, and case 2 of Definition 2 cannot happen.

Lemma 1. *Given a basic narrowing derivation $t \xrightarrow{b}_{p,l \rightarrow r, \sigma} t' \xrightarrow{b}_{q',g \rightarrow d, \theta} u$, and $q \in \text{Pos}(t)$, if q is an antecedent of q' then q and p are necessarily disjoint, q' is in the skeleton, and $q = q'$.*

Proof. Suppose that q and p are not disjoint, then we are in case 2 of Definition 2, and $q' = p.u'.w$ for some $u' \in \text{Pos}_x(r)$. But this means q' is in the environment, and hence the narrowing derivation is not basic, which contradicts the initial assumption. \square

Now we show that basic narrowing steps can be commuted under certain conditions. This result is the basis for the modularity results of Section 5.

Proposition 1 (Commutation of Basic Narrowing). *Let \mathcal{R} be a TRS and*

$$\langle s, \theta \rangle \xrightarrow{b}_{p,g \rightarrow d, \sigma_1} \langle s[d]_p, \theta \sigma_1 \rangle \xrightarrow{b}_{q,l \rightarrow r, \sigma_2} \langle s[d]_p[r]_q, \theta \sigma_1 \sigma_2 \rangle \quad (2)$$

be a sequence of two basic narrowing steps s.t. q admits an antecedent in s . Then (2) can be commuted to the following equivalent basic narrowing derivation:

$$\langle s, \theta \rangle \xrightarrow{b}_{q,l \rightarrow r, \sigma_3} \langle s[r]_q, \theta \sigma_3 \rangle \xrightarrow{b}_{p,g \rightarrow d, \sigma_4} \langle s[r]_q[d]_p, \theta \sigma_3 \sigma_4 \rangle \quad (3)$$

where $\sigma_1 \sigma_2 = \sigma_3 \sigma_4[s]$.

In order to provide the proof, we need the auxiliary notion of substitution merge [24].

Definition 4 (Unifiable Substitutions). [24] *We say that two substitutions σ and θ are unifiable iff there exists a substitution μ s.t. $\sigma\mu = \theta\mu$. The most general instance $\sigma \cap \theta = \sigma\eta = \theta\eta$ of σ and θ is called the merge of σ and θ . Note that the merge is commutative, i.e., $\sigma \cap \theta = \theta \cap \sigma$.*

The following result is established by [14], but the proof is due to Rety.

Lemma 2. [24] *Let s and t be terms unifiable with $\text{mgu}(s, t) = \sigma$ and θ be any substitution. In order to avoid conflicts of variables, suppose $D(\sigma) \subseteq \text{Var}(s) \cup \text{Var}(t)$, $D(\theta) \subseteq \text{Var}(s) \cup \text{Var}(t)$, $D(\sigma) \cap I(\sigma) = \emptyset$, $D(\theta) \cap I(\theta) = \emptyset$, and $I(\sigma) \cap I(\theta) = \emptyset$. Then*

- σ and θ are unifiable iff $s\theta$ and $t\theta$ are unifiable.
- if σ and θ are unifiable, with $\mu = \text{mgu}(s\theta, t\theta)$, then $\sigma \cap \theta = \theta\mu$.

The following schema illustrates the lemma.

$$\begin{array}{ccc} s & \xrightarrow{\sigma} & t \\ \theta \downarrow & & \downarrow \theta \\ & \xrightarrow{\mu} & \\ \mu \downarrow & & \downarrow \mu \end{array}$$

And here is the proof of Proposition 1.

Proof. To prove that derivation (3) exists, we need to show that the following conditions hold.

1. σ_3 exists and $s|_q\theta\sigma_3 = l\sigma_3$
2. σ_4 exists and $((s[r]_q)|_p)\theta\sigma_3\sigma_4 = g\sigma_4$
3. The resulting terms are equal, $s[r]_q[d]_p = s[d]_p[r]_q$, and $\sigma_1\sigma_2 = \sigma_3\sigma_4[s]$

Since $\langle s[d]_p, \theta\sigma_1 \rangle \xrightarrow{b}_{q, l \rightarrow r, \sigma_2} \langle s[d]_p[r]_q, \theta\sigma_1\sigma_2 \rangle$, then $(s[d]_p)|_q\theta\sigma_1\sigma_2 = l\sigma_2$. By Lemma 1, the antecedent of q in s is $s|_q$, hence $s|_q\theta\sigma_1 = (s[d]_p)|_q\theta\sigma_1$. On the other hand, since $D(\sigma_1) \cap \text{Var}(l) = \emptyset$ then $l\sigma_1 = l$. Lemma 2 assures σ_3 and σ_1 unify, and that σ_3 exists. This proves (1) and $\sigma_3 \cap \sigma_1 = \sigma_1\sigma_2$.

$$\begin{array}{ccc} s|_q\theta & \xrightarrow{\sigma_3} & l \\ \sigma_1 \downarrow & & \downarrow \sigma_1 \\ & \xrightarrow{\sigma_2} & \end{array}$$

Now, since $\langle s, \theta \rangle \xrightarrow{b}_{p, g \rightarrow d, \sigma_1} \langle s[d]_p, \theta\sigma_1 \rangle$ then $(s|_p)\theta\sigma_1 = g\sigma_1$. By Lemma 1, p and q are disjoint, hence $(s|_p)\theta = (s[r]_q)|_p\theta$. By considering that $D(\sigma_3) \cap \text{Var}(g) = \emptyset$ we have $g\sigma_3 = g$. Therefore, $s|_p\theta\sigma_3\sigma_4 = g\sigma_3\sigma_4$.

$$\begin{array}{ccc} s|_p\theta & \xrightarrow{\sigma_1} & g \\ \sigma_3 \downarrow & & \downarrow \sigma_3 \\ & \xrightarrow{\sigma_4} & \end{array}$$

By commutativity of substitution unification, σ_1 and σ_3 unify, showing that a σ_4 exists s.t. $\sigma_4 = \text{mgu}(s|_p\theta\sigma_3, g\sigma_3) = \text{mgu}((s[r]_q)|_p\theta\sigma_3, g)$, which proves (2). Moreover, $\sigma_3 \cap \sigma_1 = \sigma_3\sigma_4$, and hence $\sigma_1\sigma_2 = \sigma_3\sigma_4$. \square

4 Modular Termination of Basic Narrowing

Let us recall some standard notions regarding modularity of rewriting, as defined in [21], that will be used throughout the paper. Figure 1 shows diagrammatic renditions of these definitions.

disjoint $(\Sigma_0, \mathcal{R}_0)$ and $(\Sigma_1, \mathcal{R}_1)$ are *disjoint* if they do not share symbols, that is, $\Sigma_0 \cap \Sigma_1 = \emptyset$. Their union, called *direct sum*, is denoted $\mathcal{R} = \mathcal{R}_0 \uplus \mathcal{R}_1$.

constructor sharing $(\mathcal{D}_0 \uplus \mathcal{C}_0, \mathcal{R}_0)$ and $(\mathcal{D}_1 \uplus \mathcal{C}_1, \mathcal{R}_1)$ are *constructor sharing* if they do not share defined symbols, i.e., $\mathcal{D}_0 \cap \mathcal{D}_1 = \emptyset$.

composable Two systems $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$ are *composable* if $\mathcal{D}_0 \cap \mathcal{C}_1 = \mathcal{D}_1 \cap \mathcal{C}_0 = \emptyset$ and both systems share all the rewrite rules that define every shared defined symbol, i.e., $\mathcal{R}_{sh} \subseteq \mathcal{R}_0 \cap \mathcal{R}_1$ where $\mathcal{R}_{sh} = \{l \rightarrow r \in \mathcal{R}_0 \cup \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_{sh}\}$.

hierarchical combination A system $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is the *hierarchical combination* (HC) of a base system $(\mathcal{D}_0 \uplus \mathcal{C}_0, \mathcal{R}_0)$ and an extension system $(\mathcal{D}_1 \uplus \mathcal{C}_1, \mathcal{R}_1)$ iff $\mathcal{D}_0 \cap \mathcal{D}_1 = \emptyset$ and $\mathcal{C}_0 \cap \mathcal{D}_1 = \emptyset$.

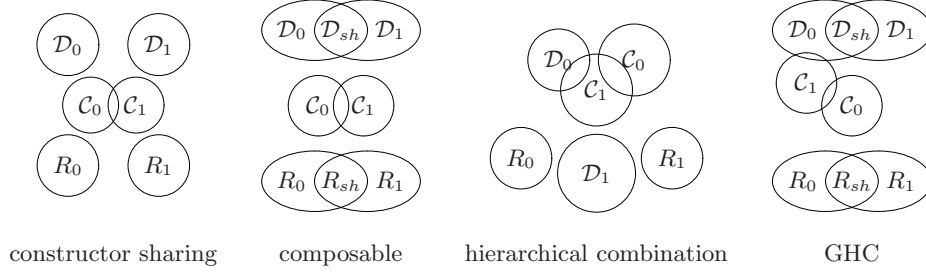


Fig. 1. Standard modular combinations

generalized hierarchical combination A system $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is the *generalized hierarchical combination* (GHC) of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and an extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$ with shared subsystem (\mathcal{F}, R_{sh}) iff $\mathcal{D}_0 \cap \mathcal{D}_1 = \emptyset$, $\mathcal{C}_0 \cap \mathcal{C}_1 = \emptyset$, $R_{sh} = R_0 \cap R_1$ where $R_{sh} = \{l \rightarrow r \in \mathcal{R}_0 \cup \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_{sh}\}$, and $\mathcal{F} = \{f \in \mathcal{F} \mid f \text{ occurs in } R_{sh}\}$.

Roughly speaking, in a hierarchical combination $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ the sets of function symbols defined in \mathcal{R}_0 and \mathcal{R}_1 are disjoint, and the defined function symbols of the base (\mathcal{R}_0) can occur in rules of the extension, but not viceversa. GHCs generalize both HCs and composable systems.

As noted by [23], this classification of combinations of TRSs is straightforwardly applicable to programming languages and incremental program development. The modularity results of direct-sums can be used when two subsystems are defined over different domains, e.g. the natural numbers and the Boolean domain. The modularity results of constructor sharing unions can be used when two subsystems define independent functions (none of the two systems use the procedures defined in the other) over a common domain. HCs model the notion of *modules* in programming languages. The following example borrowed from [21] illustrates these notions.

Example 2. Consider the following TRSs:

$$\begin{aligned}
\mathcal{R}_+ &= \begin{cases} 0 + y \rightarrow y \\ s(x) + y \rightarrow s(x + y) \end{cases} & \mathcal{R}_- &= \begin{cases} 0 - s(y) \rightarrow 0 \\ x - 0 \rightarrow x \\ s(x) - s(y) \rightarrow x - y \end{cases} \\
\mathcal{R}_* &= \begin{cases} 0 * y \rightarrow 0 \\ s(x) * y \rightarrow (x * y) + y \end{cases} & \mathcal{R}_{pow} &= \begin{cases} pow(x, 0) \rightarrow s(0) \\ pow(x, s(y)) \rightarrow x * pow(x, y) \end{cases} \\
\mathcal{R}_{app} &= \begin{cases} nil ++ ys \rightarrow ys \\ (x : xs) ++ ys \rightarrow x : (xs ++ ys) \end{cases}
\end{aligned}$$

\mathcal{R}_+ and \mathcal{R}_{app} are disjoint, \mathcal{R}_+ and \mathcal{R}_- are constructor-sharing, $\mathcal{R}_+ \cup \mathcal{R}_*$ is composable with $\mathcal{R}_+ \cup \mathcal{R}_{app}$, and $\mathcal{R}_* \cup \mathcal{R}_+$ is a HC where \mathcal{R}_* extends \mathcal{R}_+ . Lastly, the system $\mathcal{R}_1 = \mathcal{R}_{pow} \cup \mathcal{R}_+$ extends $\mathcal{R}_0 = \mathcal{R}_* \cup \mathcal{R}_+$ in a GHC with shared subsystem $\mathcal{R}_{sh} = \mathcal{R}_+$.

Note that constructor sharing systems generalize disjoint unions, and are themselves generalized by both composable and HCs. Finally, these last two notions are subsumed by GHCs.

4.1 \mathcal{C}_ε -termination

The notion of \mathcal{C}_ε -termination is used in the literature for proving termination of rewriting in a modular way [21]. The technique is essentially based on finding sufficient conditions for the modularity of the more restrictive \mathcal{C}_ε -termination property.

Definition 5 (\mathcal{C}_ε termination). [21] *A TRS \mathcal{R} is called \mathcal{C}_ε -terminating if the collapsing extended TRS $\mathcal{R} \uplus \mathcal{C}_\varepsilon$, where $\mathcal{C}_\varepsilon = \{\mathbf{Cons}(x, y) \rightarrow x, \mathbf{Cons}(x, y) \rightarrow y\}$, is terminating, where \mathbf{Cons} is a fresh symbol not occurring in the signature of \mathcal{R} .*

This definition can be lifted in the obvious way to (basic) narrowing termination of $\mathcal{R} \uplus \mathcal{C}_\varepsilon$, which we call \mathcal{C}_ε -termination of Basic Narrowing. The following consequence of Theorem 1 is interesting.

Proposition 2 (\mathcal{C}_ε -termination of Basic Narrowing). *Let \mathcal{R} be a (\xrightarrow{b}) -terminating TRS. Then, the TRS $\mathcal{R} \uplus \mathcal{C}_\varepsilon$ is also (\xrightarrow{b}) -terminating.*

Proof. Immediate by Theorem 1 because the rhs's of the rules \mathcal{C}_ε satisfy the termination condition for basic narrowing derivations of Theorem 1. \square

In contrast, the corresponding property does not generally hold for (\rightarrow) -termination, but only for termination of innermost rewriting (see [21]).

4.2 Constructor-sharing and composable unions

The following result is a direct consequence of Theorem 1.

Theorem 2 (Modularity of constructor-sharing unions). *Termination of basic narrowing is modular for constructor-sharing systems.*

Proof. Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be a constructor-sharing union of two (\xrightarrow{b}) -terminating TRSs. We need to prove that any basic narrowing derivation issuing from every right hand side in \mathcal{R} terminates. But this is easy, because every derivation stemming from the right hand side of a rule of \mathcal{R}_0 only use rules from \mathcal{R}_0 (there is no redex w.r.t. \mathcal{R}_1), and then are finite by hypothesis since \mathcal{R}_0 is (\xrightarrow{b}) -terminating. The same argument applies to the right hand sides of the rules of \mathcal{R}_1 . By Theorem 1, the conclusion follows. \square

This implies modularity for disjoint unions too, as in the following well-known example.

Example 3 (Toyama). Let us consider Toyama's example [26]:

$$\mathcal{R}_0 : f(0, 1, x) \rightarrow f(x, x, x) \qquad \mathcal{R}_1 : g(x, y) \rightarrow x \quad g(x, y) \rightarrow y$$

Basic narrowing trivially terminates on each system, since every rhs is clearly unnarrowable. By Theorem 2, it also terminates for $\mathcal{R}_0 \cup \mathcal{R}_1$.

It is well known that the system $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is not (\rightarrow) -terminating. However, it is innermost terminating. This shows that $(\overset{b}{\rightarrow})$ -termination does not entail (\rightarrow) -termination, which suggests that the modularity requirements for $(\overset{b}{\rightarrow})$ -termination are less restrictive than those of (\rightarrow) -termination. Actually, the modularity properties of $(\overset{b}{\rightarrow})$ -termination are comparable to those of innermost (\rightarrow) -termination (see e.g. [21]). The next theorem extends the modularity of $(\overset{b}{\rightarrow})$ -termination to composable systems.

Theorem 3 (Modularity of composable unions). *Termination of basic narrowing is a modular property of composable systems.*

Proof. Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be the union of two composable, $(\overset{b}{\rightarrow})$ -terminating TRSs. We can partition the rules in \mathcal{R} in the following disjoint sets:

$$\begin{aligned} \mathcal{R}_{sh} &= \mathcal{R}_0 \cap \mathcal{R}_1 \\ \mathcal{R}'_0 &= \mathcal{R}_0 \setminus \mathcal{R}_1 \\ \mathcal{R}'_1 &= \mathcal{R}_1 \setminus \mathcal{R}_0 \end{aligned}$$

Basic narrowing derivations issuing from the right hand sides of the rules in \mathcal{R}'_0 or \mathcal{R}'_1 terminate by an argument analog to the one used in the proof of Proposition 2. It is easy to see that derivations issuing from the right hand sides of \mathcal{R}_{sh} are terminating too, since they include no redex w.r.t. either \mathcal{R}'_0 or \mathcal{R}'_1 , and basic narrowing does never reduce any redex occurring at a non-basic position (i.e. in the environment part of the tuple). By Theorem 1, the conclusion follows. \square

In Section 4.5 we prove that termination of basic narrowing is modular for proper extensions (PE) [23], a fairly general restriction of HCs. To achieve this result, we proceed as follows. In Section 4.4 we prove the modularity for *nice* extensions (NE) [23], a restriction of PEs. Then, we apply a result from [23] that relates NEs to PEs, which delivers the desired result.

In the remaining of this section we make use of the following notion.

Definition 6 (Dependency relation $\triangleright_{\mathcal{R}}$). [23] *For a TRS $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ the dependency relation $\triangleright_{\mathcal{R}}$ is the smallest preorder satisfying the condition $f \triangleright_{\mathcal{R}} g$ whenever there is a rewrite rule $f(s_1, \dots, s_n) \rightarrow r \in \mathcal{R}$ and $g(t_1, \dots, t_n)$ is a subterm of r , with $g \in \mathcal{D}$.*

We often omit \mathcal{R} from $\triangleright_{\mathcal{R}}$ when it is clear from the context. We say that a symbol $f \in \mathcal{D}$ depends on a symbol $g \in \mathcal{D}$ if $f \triangleright g$. Intuitively, $f \triangleright g$ if the evaluation of f involves a call to g after one or more rewrite steps.

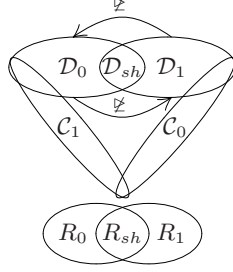


Fig. 2. Crosswise independent union

4.3 Crosswise independent unions

Crosswise independent unions (CIU) are a generalization of composable unions in which the non-shared defined symbols are allowed to appear in the left hand sides of rules (but not in the right hand sides). Figure 2 pictorially renders this class of unions. We study them mainly because they are useful for the proof of GNEs.

Definition 7 (Crosswise independent union). [23] *Two TRS's $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$ are crosswise independent if (i) $\mathcal{R}_{sh} = \mathcal{R}_0 \cap \mathcal{R}_1 = \{l \rightarrow r \in \mathcal{R}_0 \cup \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_{sh}\}$ and (ii) for all $f \in \mathcal{D}_i \cup \mathcal{D}_{sh}$ and $g \in \mathcal{D}_{1-i}$, $i \in \{0, 1\}$, we have $f \not\vdash_{\mathcal{R}_0 \cup \mathcal{R}_1} g$. We say that $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is a crosswise independent union if \mathcal{R}_0 and \mathcal{R}_1 are crosswise independent.*

We show that (\xrightarrow{b}) -termination is modular for CIUs, as an instrumental result for proving the modularity of PEs.

Theorem 4. *Termination of basic narrowing is modular for crosswise independent unions.*

Proof. Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be the union of two crosswise independent systems. The scheme of the proof is similar to the proof of Theorem 3. Essentially, since all the derivations starting from the rhs of a rule in $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ only use rules that belong either to \mathcal{R}_0 or \mathcal{R}_1 , and are hence finite in both cases, by Theorem 1 every basic narrowing derivation w.r.t. \mathcal{R} is finite. \square

4.4 Nice extensions

Nice extensions (NE) are a restriction of PEs introduced by Krishna Rao [23]. NEs are a useful intermediate notion, because it can be shown that every PE can be modelled as a *pyramid* of NEs, which we do in Section 4.5.

Definition 8 (Split). *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. The set \mathcal{D}_1 of defined symbols*

of \mathcal{R}_1 is split in two sets \mathcal{D}_1^0 and \mathcal{D}_1^1 where \mathcal{D}_1^0 contains all the symbols that depend on function symbols from \mathcal{R}_0 , i.e., $\mathcal{D}_1^0 = \{f \in \mathcal{D}_1 \mid \exists g \in \mathcal{D}_0, f \triangleright_{\mathcal{R}} g\}$ and $\mathcal{D}_1^1 = \mathcal{D}_1 \setminus \mathcal{D}_1^0$. We can then split \mathcal{R}_1 in two subsystems \mathcal{R}_1^0 and \mathcal{R}_1^1 as $\mathcal{R}_1^0 = \{l \rightarrow r \in \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_1^0\}$ and $\mathcal{R}_1^1 = \{l \rightarrow r \in \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_1^1\}$.

Definition 9 (Generalized Nice extension). [23] Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be the GHC of the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$ over the base $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$. \mathcal{R}_1 is a generalized nice extension (GNE) of \mathcal{R}_0 if, for every rewrite rule $l \rightarrow r \in \mathcal{R}_1$, and for every subterm s of r such that $\text{root}(s) \in \mathcal{D}_1^0$, s contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.

An Example of NE can be found in Example 4.

We identify a special set $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ of terms that represent the right hand sides of rules of the TRSs that can be obtained as GNEs. This allows us to prove that basic narrowing w.r.t. $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ terminates only if it terminates for the terms in $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$. Let us introduce the standard notion of context here. A context is a term C with zero or more ‘holes’, i.e., the fresh constant symbol \square . If C is a context and \bar{t} a list of terms, $C[\bar{t}]$ denotes the result of replacing the holes in C by the terms in \bar{t} .

Definition 10 ($\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terms). Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a generalized nice combination of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. Let \mathcal{CC}_{01} be the set of contexts of $(\mathcal{C} \cup \mathcal{D}_0 \cup \mathcal{D}_{sh} \cup \mathcal{D}_1^1)$. We define $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ as the set of all terms of the form $C[s_1, \dots, s_n]$, where $C \in \mathcal{CC}_{01}$ and the following conditions hold:

1. for all i , $\text{root}(s_i) \in \mathcal{D}_1^0$.
2. s_i contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.

By definition, $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terms have the property that no \mathcal{R}_1^0 reduction step is possible within the context C . Also, the set $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ is closed under $\xrightarrow{b}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ if $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is a generalized nice combination.

The reader can check that the right hand sides of the rules in a nice combination fulfill the conditions above. In order to prove the (\xrightarrow{b}) -termination of a system, by Theorem 1 it suffices to prove that derivations starting from the right hand sides of the rules are finite. We prove the more general result that derivations starting from $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terms are finite. This is done in Section 5 for a superset of $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$, via a commutation result of abstract reduction systems (ARS) given in Theorem 6 (c.f. Section 5).

Corollary 1. Let \mathcal{R}_1 be a GNE over \mathcal{R}_0 . Every basic narrowing derivation in $\mathcal{R}_0 \cup \mathcal{R}_1$ starting from a term of $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terminates.

Now we can easily generalize this result to any term by applying Theorem 1.

Corollary 2. Termination of basic narrowing is modular for generalized nice extensions.

4.5 Proper extensions

In this section, we extend our previous modularity results from NEs to PEs, by reusing a result from Krishna Rao that relates proper and generalized nice extensions.

Definition 11 (Generalized Proper extension). [23] *Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. \mathcal{R}_1 is a generalized proper extension (GPE) of \mathcal{R}_0 if each rewrite rule $l \rightarrow r \in \mathcal{R}_1^0$ satisfies that, for every subterm t of r such that $\text{root}(t) \in \mathcal{D}_1^0$ and $\text{root}(t) \succeq_{\mathcal{R}} \text{root}(l)$, t contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

Example 4. Consider computing the factorial of a number in tail recursive style.

$$\mathcal{R}_! = \begin{cases} \text{fact}(x) \rightarrow \text{factacc}(x, 1) \\ \text{factacc}(0, x) \rightarrow x \\ \text{factacc}(\text{s}(y), x) \rightarrow \text{factacc}(y, x * \text{s}(y)) \end{cases} \quad \mathcal{R}_* = \begin{cases} 0 * y \rightarrow 0 \\ \text{s}(x) * y \rightarrow (x * y) + y \end{cases}$$

$\mathcal{R}_!$ is a hierarchical extension of \mathcal{R}_* , but it is not a PE (because of the 3rd rule). On the other hand, the standard, non tail recursive presentation of factorial is a PE, and moreover a NE.

To understand why non proper extensions can be troublesome for termination, consider the following example.

Example 5. Consider the following TRSs, whose combination is hierarchical but not proper:

$$\mathcal{R}_1 : \{\mathbf{f}(\mathbf{a}) \rightarrow \mathbf{f}(\mathbf{b})\} \quad \mathcal{R}_0 : \{\mathbf{b} \rightarrow \mathbf{a}\}$$

There exists the following infinite basic narrowing derivation

$$\mathbf{f}(\mathbf{a}) \xrightarrow{\mathbf{b}} \mathbf{f}(\mathbf{b}) \xrightarrow{\mathbf{b}} \mathbf{f}(\mathbf{a}) \xrightarrow{\mathbf{b}} \dots$$

produced by the nesting of a redex w.r.t. \mathcal{R}_0 inside the recursive call to \mathbf{f} in the rhs of the rule of \mathcal{R}_1 .

PEs are less restrictive than NEs because they allow nesting of \mathcal{R}_0 functions only as long as they do not occur inside a recursive definition, whereas NEs forbid any function nesting. That is, every NE is also a PE, but not the other way around. As stated before, we can model any GPE as a finite pyramid of one or more GNEs. Essentially, the idea is similar to the modular decomposition of a TRS in [27]. What we do is to reduce a given PE to the canonical modular form, a modular partition such that each of the individual modules cannot be split up. In order to achieve this we employ the graph induced by the dependency relation \succeq on defined function symbols, and the rules corresponding to the symbols of every strongly connected component become a module (i.e., a GNE).

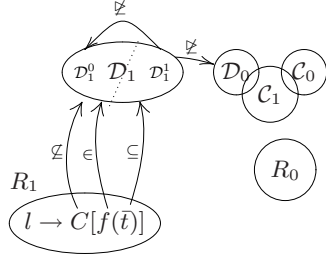


Fig. 3. Nice extension

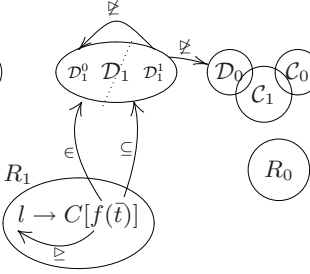


Fig. 4. Proper extension

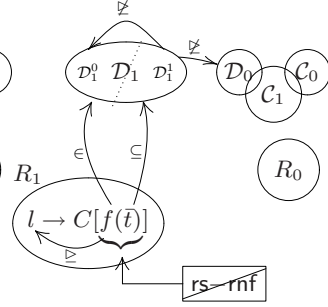


Fig. 5. Relaxed Proper extension

Lemma 3. [23] *Let \mathcal{R}_1 be a finite TRS s.t. it is a GPE of \mathcal{R}_0 . \mathcal{R}_1 can be seen as a finite pyramid of GNEs.*

We are now ready to give our final modularity result for termination of basic narrowing in GPEs, which follows from the previous lemma and Corollary 2.

Corollary 3. *Termination of basic narrowing is modular for generalized proper extensions.*

In the last section of the paper, by weakening some conditions of GPEs, we provide a novel class of composition of TRSs called *relaxed proper extensions* for which the modularity of basic narrowing termination still holds.

5 Relaxed proper extensions

Let us introduce the main idea behind our generalization of GPEs by means of the following example.

Example 6. Consider the following TRS, an encoding³ of the exponentiation x^y and the exclusive or operators that are commonly used in the specification of many cryptographic protocols. [7, 8], where the constructor symbol g is used as a generator for the exponentiation.

$$\begin{aligned} \mathcal{R}_1 : \exp(\exp(g, X), Y) &\rightarrow \exp(g, X * Y) \\ \mathcal{R}_0 : X * X^{-1} &\rightarrow 1 \quad X * 1 \rightarrow X \quad 1 * X \rightarrow X \end{aligned}$$

Basic narrowing trivially terminates on each system separately, since every rhs is clearly unarrowable. However, their combination $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is not a PE, although it is easy to see that basic narrowing indeed terminates in \mathcal{R} . The reason

³ We are aware that this encoding is not complete since the exclusive or operator is associative and commutative; nevertheless, the example is useful for motivation.

is that the outer function symbol \mathbf{exp} in the recursive invocation occurring in the right hand side of the rule of \mathcal{R}_1 is blocked forever, and therefore cannot cause nontermination. The following novel notion of *relaxed* proper extension (RPE) captures this idea.

We introduce the notion of *root-stable rigid normal form*, which lifts to narrowing the standard concept of head normal form. By abusing notation, we apply this notion, with no change, to basic narrowing.

Definition 12 (root-stable rigid normal form). [2] *A term s is a root-stable rigid normal form (rs-rnf) w.r.t. \mathcal{R} if either s is a variable or there are no substitutions θ and θ' and terms s' and s'' s.t. $s\theta \xrightarrow{\epsilon^*}_{\mathcal{R}} s' \xrightarrow{b}_{\epsilon, \mathcal{R}, \theta'} s''$.*

Definition 13 (Generalized Relaxed proper extension). *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. \mathcal{R}_1 is a generalized relaxed proper extension (GRPE) of \mathcal{R}_0 if every rule in \mathcal{R}_1^0 satisfies the following condition:*

(H1) *for each subterm t of r such that (a) $\text{root}(t) \in \mathcal{D}_1^0$, (b) t is not a rs-rnf, and (c) $\text{root}(t) \succeq_{\mathcal{R}} \text{root}(l)$, t does not contain a function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

The reader can check that the TRS of Example 6 is indeed a GRPE. In the following, we show that (\sim^b) -termination is modular for RPEs by showing first its modularity for GRNEs, and then establishing a relation between GRPEs and GRNEs. The reasoning is similar to the one followed in Section 4.5.

Definition 14 (Generalized Relaxed nice extension). *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. \mathcal{R}_1 is a generalized relaxed nice extension (GRNE) of \mathcal{R}_0 if it is a GRPE, and for every rewrite rule $l \rightarrow r \in \mathcal{R}_1$ the following condition holds:*

(N1) *for each subterm t of r such that t is not a rs-rnf and $\text{root}(t) \in \mathcal{D}_1^0$, t contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

We can extend $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ to precisely capture the right hand sides of GRNEs.

Definition 15 ($\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$ terms). *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GRNE of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. We define $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$ as the set of all terms of the form $C[s_1, \dots, s_n]$, where C is a context in $(\mathcal{D} \cup \mathcal{C})$ and the following conditions hold:*

1. *no reduction is possible in \mathcal{R}_1^0 at a position within the context C .*
2. *for all i , $\text{root}(s_i) \in \mathcal{D}_1^0$.*
3. *for all i , s_i is not a rs-rnf.*

4. s_i contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.

Note that $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}} \subseteq \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$. The set $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$ also enjoys the property of being closed under $\xrightarrow{b}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ in a GRNE.

Lemma 4. *If $t \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$ and either $t \xrightarrow{b}_{\mathcal{R}_1, \theta} t'$ or $t \xrightarrow{b}_{\mathcal{R}_0, \theta} t'$, then $t' \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$.*

Proof. First we consider the case for \mathcal{R}_1 . By definition $t = C[s_1, \dots, s_n]$, and let $l \rightarrow r \in \mathcal{R}_1$ be the rule applied in the reduction step. We consider two cases depending on whether $\text{root}(l)$ is in \mathcal{D}_1^0 or not.

- (a) $\text{root}(l) \notin \mathcal{D}_1^0$. That is, $\text{root}(l) \in (\mathcal{D}_1^1 \cup \mathcal{D}_{sh})$. There are two subcases.
 - (1) The reduction took place in C . By definition, no \mathcal{D}_1^0 symbol occurs in r , hence t' is of the form $C'[t_1, \dots, t_m]$, where $C' \in \mathcal{CC}_{01}$, $\text{root}(t_i) \in \mathcal{D}_1^0$ and each t_i is a subterm of some s_j . The lemma holds.
 - (2) The reduction took place in a proper subterm of some s_i and t' is of the form $C[s_1, \dots, s'_j, \dots, s_n]$. As in case (1), since no \mathcal{D}_1^0 symbol occurs in r , the lemma holds.
- (b) $\text{root}(l) \in \mathcal{D}_1^0$. Therefore, the reduction happened at the root of some s_i , and r is a term in $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$, $r = C'[u_1, \dots, u_m]$. t' is of the form $C[s_1, \dots, s_{i-1}, r, s_{i+1}, \dots, s_n] = C''[s_1, \dots, s_{i-1}, u_1, \dots, u_m, s_{i+1}, \dots, s_n]$, where C'' is the context resulting of joining C and C' .

Next, we consider the closedness for \mathcal{R}_0 . By definition $t = C[s_1, \dots, s_n]$, and let $l \rightarrow r \in \mathcal{R}_0$ be the rule applied in the reduction step. The reduction takes place in C , and since no symbol in \mathcal{D}_1 occurs in r . By definition we have that t' is of the form $C'[t_1, \dots, t_m]$ where $C' \in \mathcal{CC}_{01}$, $\text{root}(t_i) \in \mathcal{D}_1^0$ and each t_i is a subterm of some s_j . The result follows. \square

The rest of this section is devoted to extending Corollary 1 to the set $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$. First, let us recall some general results on quasi-commutation of abstract relations.

Definition 16 (Abstract Reduction System). *An abstract reduction system (ARS) is a structure $\mathcal{A} = (A, \{\rightarrow_\alpha \mid \alpha \in I\})$ consisting of a set A and a set of binary relations \rightarrow_α on A , indexed by a set I . We write $(A, \rightarrow_1, \rightarrow_2)$ instead of $(A, \{\rightarrow_\alpha \mid \alpha \in \{1, 2\}\})$.*

Definition 17 (Quasi-commutation). [6] *Let \rightarrow_0 and \rightarrow_1 be two relations on a set S . The relation \rightarrow_1 quasi-commutes over \rightarrow_0 if, for all $s, u, t \in S$ s.t. $s \rightarrow_0 u \rightarrow_1 t$, there exists $v \in S$ s.t. $s \rightarrow_1 v \rightarrow_{01}^* t$, where \rightarrow_{01}^* is the transitive-reflexive closure of $\rightarrow_0 \cup \rightarrow_1$.*

Theorem 5. [6] *If the relations \rightarrow_0 and \rightarrow_1 in the ARS $(S, \rightarrow_0, \rightarrow_1)$ are strongly normalizing and \rightarrow_1 quasi-commutes over \rightarrow_0 , the relation $\rightarrow_0 \cup \rightarrow_1$ is strongly normalizing too.*

We now define an ARS with skeleton–environment pairs as elements, where the skeletons come from the set $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$ of terms, and the relationships \rightarrow_0 and \rightarrow_1 of the ARS are restrictions of basic narrowing.

Definition 18. Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be a generalized relaxed nice combination. We define the ARS $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1) = (\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}} \times \text{Subst}, \rightarrow_0, \rightarrow_1)$, where the relations \rightarrow_0 and \rightarrow_1 are defined as follows. Let $s = C[s_0, \dots, s_n]$ be a term in $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$. Then

1. $\langle C[s_0, \dots, s_n], \sigma \rangle \rightarrow_0 \langle C'[s_0, \dots, s_n], \theta \sigma \rangle$ if $\langle C[s_0, \dots, s_n], \sigma \rangle \xrightarrow{b}_{\mathcal{R}_0 \cup \mathcal{R}_1, \theta} \langle C'[s_0, \dots, s_n], \sigma \theta \rangle$ is a basic narrowing step given within the context C
2. $\langle C[s_0, \dots, s_n], \sigma \rangle \rightarrow_1 \langle C[s_0, \dots, s_{i-1}, s'_i, s_{i+1}, \dots, s_n], \theta \sigma \rangle$ if $\langle C[s_0, \dots, s_n], \sigma \rangle \xrightarrow{b}_{\mathcal{R}_1, \theta} \langle C[s_0, \dots, s_{i-1}, s'_i, s_{i+1}, \dots, s_n], \theta \sigma \rangle$ is a basic narrowing step given at a subterm s_i , with $i \in [0, \dots, n]$.

The relation $\rightarrow_1 \cup \rightarrow_0$ is exactly the basic narrowing relation over $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$. In the following we establish that both \rightarrow_0 and \rightarrow_1 are terminating relations.

Lemma 5. Given the ARS $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1)$ of Definition 18, the relations \rightarrow_0 and \rightarrow_1 are terminating if \mathcal{R}_0 and \mathcal{R}_1 are (\xrightarrow{b}) -terminating.

Proof. The relation \rightarrow_1 is a subrelation of $\xrightarrow{b}_{\mathcal{R}_1}$, and hence terminating.

Now consider the subsystem $\mathcal{R}'_1 = \mathcal{R}_1^1 \cup \mathcal{R}'$. It can be shown that \mathcal{R}_0 and \mathcal{R}'_1 are crosswise independent and hence their union is terminating by Theorem 4. From the definition of \mathcal{A} , it can be seen that \rightarrow_0 is actually a restriction of this system to the terms in $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}$, and therefore termination of \rightarrow_0 follows. \square

We are now in a position to prove the quasi-commutation of the relation \rightarrow_1 over the relation \rightarrow_0 in the ARS $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1)$. The proof of this result relies on Proposition 1.

Theorem 6. Given the ARS $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1)$ of Definition 18, the relation \rightarrow_1 quasi-commutes over the relation \rightarrow_0 .

Proof. We have to show that

$$\forall s, u, t \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}} \text{ s.t. } s \xrightarrow{p}_0 u \xrightarrow{q}_1 t, \exists v \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}}, p' \in \text{Pos}_\Sigma(s) \text{ s.t. } s \xrightarrow{p'}_1 v \xrightarrow{*}_{01} t \quad (4)$$

where, by abusing notation, p, q and p' refer to positions that are in the skeleton part of the tuples denoted by s, u, t and v .

As $\rightarrow_0 \subseteq \rightarrow_{01}$ and $\rightarrow_1 \subseteq \rightarrow_{01}$, we can equivalently see these as \rightarrow_{01} derivations. Since \rightarrow_{01} is a basic narrowing relation, we can use Proposition 1 to prove the result. We only need to show that q admits an antecedent in s . It is easy to see that this is satisfied, because \rightarrow_0 cannot create new \rightarrow_1 redexes, which means that redex $u|_q$ was already there in $s|_q$. On the other hand, since no redex can be propagated by basic narrowing, p and q must be disjoint. Applying Proposition 1, we have:

$$\forall s, u, t \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}} \text{ s.t. } s \xrightarrow{p}_{01} u \xrightarrow{q}_{01} t, \exists v \in \mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{\text{rs-rnf}} \text{ s.t. } s \xrightarrow{q}_{01} v \xrightarrow{p}_{01} t \quad (5)$$

from which we observe that (1) $p' = q$, (2) t is reached in a single \rightarrow_0 step, and (3) the step $s \rightarrow_{01} t$ is indeed given in \rightarrow_1 , which suffices to prove the result. \square

Then, as a straightforward consequence of Theorem 5 and Theorem 6, we derive the relaxed version of Corollary 1.

Corollary 4. *Let \mathcal{R}_1 be a GRNE over \mathcal{R}_0 . Every basic narrowing derivation in $\mathcal{R}_0 \cup \mathcal{R}_1$ starting from a term of $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}^{rs-nf}$ terminates.*

By Theorem 1, we obtain the desired modularity result for basic narrowing in our generalization of GNEs.

Corollary 5. *Termination of basic narrowing is modular for generalized relaxed nice extensions.*

We now study the connection between GRNEs and GRPEs, and extend the results and proofs from [23] extending them to our generalized relaxed nice extensions.

Lemma 6. *Let \mathcal{R}_1 be a finite TRS s.t. it is a GRPE of \mathcal{R}_0 . \mathcal{R}_1 can be seen as a finite pyramid of GRNEs.*

In order to prove this Lemma, we need first two auxiliary definitions.

Definition 19 (Equivalence relation \approx and partial order \sqsupset). [23] *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a generalized hierarchical combination of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. We define an equivalence relation $\approx_{\mathcal{R}}$ from the dependency relation $\supseteq_{\mathcal{R}}$, where the equivalence class containing f is denoted by $[f]_{\mathcal{R}}$, and a partial ordering $\sqsupset_{\mathcal{R}}$ on the set of equivalence classes:*

- $f \approx_{\mathcal{R}} g$ iff $f \supseteq_{\mathcal{R}} g$ and $g \supseteq_{\mathcal{R}} f$, where $f, g \in \mathcal{D}_1^0$.
- $[f]_{\mathcal{R}} \sqsupset_{\mathcal{R}} [g]_{\mathcal{R}}$ iff $f \supseteq_{\mathcal{R}} g$ and $g \not\supseteq_{\mathcal{R}} f$.

Since the signature of any TRS is a countable set, the equivalence relation \approx partitions \mathcal{D}_1^0 into a countable set E of equivalence classes. If we can assume that the ordering \sqsupset is noetherian, then it can be extended to a well-ordering of type λ , where λ is a countable ordinal.

Definition 20. [23] *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. For any ordinal α we denote the α -th element in the above well-ordering by E_{α} (if $\alpha > \lambda$ then $E_{\alpha} = \emptyset$). We define the TRS $X_{\alpha} = \{l \rightarrow r \in \mathcal{R}_1 \mid \text{root}(l) \in \mathcal{D}_{sh} \cup \mathcal{D}_1^1 \cup E_{\alpha}\}$, and the combined system $S_{\alpha} = \mathcal{R}_0 \cup (\bigcup_{\beta < \alpha} X_{\beta})$. So S_0 is \mathcal{R}_0 and S_k for $k > \lambda$ is $\mathcal{R}_0 \cup \mathcal{R}_1$.*

Theorem 7. *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be a GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and the extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets \mathcal{D}_1^0 , \mathcal{D}_1^1 , \mathcal{R}_1^0 and \mathcal{R}_1^1 as in Definition 8. If the relation $\sqsupset_{\mathcal{R}}$ is noetherian, then X_{α} is a GRNE of S_{α} for every ordinal α , where S_{α} and X_{α} are defined as in Definition 20.*

Proof. For every α , we have that S_α and X_α form a GHC. We have to show that (N1) holds in X_α . That is, for every $l \rightarrow r \in X_\alpha$, if s is a subterm of r s.t. $\text{root}(s) \in E_\alpha$ and some subterm of s contains a defined symbol depending on $\text{Def}(S_\alpha) - \text{Def}(X_\alpha)$, then s must be a **rs-rnf**. Since $\text{root}(s) \in E_\alpha$, it follows that $\text{root}(l) \in E_\alpha$ and, by definition, $\text{root}(l) \in \mathcal{D}_1^0$ and $\text{root}(s) \approx \text{root}(l)$. Therefore $\text{root}(s) \supseteq \text{root}(l)$. Now, since \mathcal{R}_1 is a GRPE, the following holds by (H1): if u contains a defined symbol depending on \mathcal{D}_0 , then u must be a **rs-rnf**. Finally, since $\text{Def}(S_\alpha) - \text{Def}(X_\alpha) \subset \mathcal{D}_0 \cup \mathcal{D}_1^0$, X_α is a GRNE of S_α . \square

And now we can prove Lemma 6.

Proof. By Theorem 7, if we can assume that the relation $\sqsubset_{\mathcal{R}_0 \cup \mathcal{R}_1}$ is noetherian. \square

Finally, we are able to establish the most general result of the paper, which follows directly from Corollary 5 and Lemma 6.

Corollary 6. *Termination of basic narrowing is modular for generalized relaxed proper extensions.*

6 Conclusions

The completeness and termination properties of basic narrowing have been studied previously in landmark work [16, 24, 20]. Recently we have contributed to the study of narrowing termination based on the termination of basic narrowing in [2]. In this paper, we improve our characterization of basic narrowing termination by proving modular termination in several hierarchical combinations of TRSs, including generalized proper extensions with shared subsystem.

Our main motivation for this work is proving termination of narrowing [16, 2]. Narrowing has received much attention due to the different applications, such as automated proofs of termination [5], execution of functional–logic programming languages [13, 18], symbolic reachability [19], verification of cryptographic protocols [10], equational unification [16], equational constraint solving [3, 4], and model checking [11], among others. Termination of narrowing is, therefore, of much interest to these applications.

References

1. M. Alpuente, S. Escobar, and J. Iborra. Modular termination of basic narrowing. Technical Report DSIC-II/04/08, Universidad Politcnica de Valencia, 2007.
2. M. Alpuente, S. Escobar, and J. Iborra. Termination of Narrowing revisited. *Theoretical Computer Science*, 2008, to appear.
3. M. Alpuente, M. Falaschi, M. Gabbrielli, and G. Levi. The semantics of equational logic programming as an instance of CLP. In *Logic Programming Languages:49–81*. The MIT Press, 1993.
4. M. Alpuente, M. Falaschi, and G. Levi. Incremental Constraint Satisfaction for Equational Logic Programming. *Theoretical Computer Science*, 142(1):27–57, 1995.

5. Thomas Arts and Jürgen Giesl. Termination of term rewriting using dependency pairs. *Theor. Comput. Sci.*, 236(1-2):133–178, 2000.
6. L Bachmair and N Dershowitz. Commutation, transformation, and termination. *Proc. of the 8th Int’l Conf. on Automated Deduction:5-20*, Jan 1986.
7. H. Comon-Lundh. Intruder Theories (Ongoing Work). In *Proc. FoSSaCS 2004*, pages 1–4. Springer LNCS 2987, 2004.
8. V. Cortier, S. Delaune, and P. Lafourcade. A Survey of Algebraic Properties used in Cryptographic Protocols. *Journal of Computer Security*, 14(1):1–43, 2006.
9. Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In *Handbook of Theoretical Computer Science vol. B:244–320*. Elsevier Science, 1990.
10. S. Escobar, C. Meadows, and J. Meseguer. A Rewriting-Based Inference System for the NRL Protocol Analyzer and its Meta-Logical Properties. *TCS 367*, 2006.
11. Santiago Escobar and José Meseguer. Symbolic model checking of infinite-state systems using narrowing. In *Proc. RTA 2007*, pages 153–168. LNCS 4533, 2007.
12. M. Fay. First-Order Unification in an Equational Theory. In *Fourth Int’l Conf. on Automated Deduction*, pages 161–167, 1979.
13. M. Hanus. The Integration of Functions into Logic Programming: From Theory to Practice. *Journal of Logic Programming*, 19&20:583–628, 1994.
14. A. Herold. Some basic notions of first-order unification theory. Technical Report SR-83-15, SEKI report, 1983.
15. S. Hölldobler. *Foundations of Equational Logic Programming*, volume 353 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlin, 1989.
16. J.-M. Hullot. Canonical Forms and Unification. In *Proc. of the 5th Int’l Conf. on Automated Deduction*, LNCS 87, pages 318–334, Berlin, 1980. Springer-Verlag.
17. J.-M. Hullot. *Compilation de Formes Canoniques dans les Théories éuationnelles. Thèse de Doctorat de Troisième Cycle*. PhD thesis, Université de Paris Sud, Orsay (France), 1981.
18. J. Meseguer. Multiparadigm logic programming. In H. Kirchner and G. Levi, editors, *Proc. ALP’92*, LNCS 632:158–200, Berlin, 1992. Springer-Verlag.
19. José Meseguer and Prasanna Thati. Symbolic reachability analysis using narrowing and its application to verification of cryptographic protocols. *HOSC:123–160*, 2007.
20. A. Middeldorp and E. Hamoen. Completeness Results for Basic Narrowing. *J. of Applicable Algebra in Engineering, Comm. and Computing*, 5:313–353, 1994.
21. E. Ohlebusch. *Advanced Topics in Term Rewriting*. Springer-Verlag, 2002.
22. C. Prehofer. On Modularity in Term Rewriting and Narrowing. In *CCL’94*, volume 845 of *Lecture Notes in Computer Science*, pages 253–268. Springer, 1994.
23. M.R.K. Krishna Rao. Modular proofs for completeness of hierarchical term rewriting systems. *Theoretical Computer Science*, Jan 1995.
24. Pierre Réty. Improving Basic Narrowing Techniques. In *RTA*. LNCS 256, 1987.
25. TeReSe, editor. *Term Rewriting Systems*. Cambridge University Press, 2003.
26. Yoshihito Toyama. Counterexamples to termination for the direct sum of term rewriting systems. *Inf. Process. Lett.*, 25(3):141–143, 1987.
27. Xavier Urbain. Modular & incremental automated termination proofs. *Int. J. Approx. Reasoning*, 32(4):315–355, 2004.