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 [15], hence termination results for narrowing have been traditionally achieved as a by–product of addressing the decidability of equational unification. Basic narrowing [15] 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 [15], 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 [16], 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 [15] is to require that the TRS satisfies Réty's maximal commutation condition for narrowing sequences [23], as we proved[1] 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, the UPV grant 3249 PAID0607 and the UPV grant FPI-UPV 2006-01.

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

A. Voronkov (Ed.): RTA 2008, LNCS 5117, pp. 1–16, 2008.

termination of basic narrowing implies termination of narrowing, namely right-linear TRSs that are either left-linear or regular and where narrowing computes[2] 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 [21], 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 [21] 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 [21], 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 4.3 that termination of basic narrowing is modular for proper extensions [22], we first prove an intermediate result: in Section 4.2 we prove that basic narrowing termination is modular for a restriction of proper extensions called *nice* extensions [22]. 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 [9,20,24].

$\mathcal{V}$ 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. Two positions p, q are disjoint, denoted by $p \parallel q$, if neither $p < q$, $p > q$, nor $p = q$. Given $S \subseteq \Sigma \cup \mathcal{V}$, $\mathcal{P}os_S(t)$ denotes the set of positions of a term t that are rooted by function symbols or variables in S. $\mathcal{P}os_{\{f\}}(t)$ with $f \in \Sigma \cup \mathcal{V}$ will be simply denoted by $\mathcal{P}os_f(t)$, and $\mathcal{P}os_{\Sigma\cup\mathcal{V}}(t)$ will be simply denoted by $\mathcal{P}os(t)$. $t|_p$ 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 $\mathit{Var}(s)$, 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 $\{x_1/t_1, \ldots, x_n/t_n\}$ for variables $x_1, \ldots, x_n$ and terms $t_1, \ldots, t_n$. 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 $\theta_{\restriction \mathit{Var}(s)}$ to denote the restriction of the substitution θ to the set of variables in s; by abuse of notation, we often simply write $\theta_{\restriction s}$. Given a term t, $\theta = \nu\ [t]$ iff $\theta_{\restriction \mathit{Var}(t)} = \nu_{\restriction \mathit{Var}(t)}$, that is, $\forall x \in \mathit{Var}(t)$, $x\theta = x\nu$. 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 $mgu(s, t)$, 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 $l \to r$ such that $l, r \in \mathcal{T}(\Sigma, \mathcal{V})$, $l \notin \mathcal{V}$, and $\mathit{Var}(r) \subseteq \mathit{Var}(l)$. 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 $\mathcal{D} = \{f \mid f(t_1, \ldots, t_n) \to r \in R\}$ 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 $\mathit{Def}(\mathcal{R})$ 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 $p \in \mathcal{P}os_\Sigma(s)$, $l \to r \in \mathcal{R}$, 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 $\to_{\mathcal{R}}$. We also omit the reduced position p when it is not relevant. A term s is a *normal form* w.r.t. the relation $\to_{\mathcal{R}}$ (or simply a normal form), if there is no term t such that $s \to_{\mathcal{R}} t$. 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 $t, t' s.t.\ s \to^*_{\mathcal{R}} t' \xrightarrow{\epsilon}_{\mathcal{R}} t$. A TRS $\mathcal{R}$ is *($\to$)-terminating* (also called strongly normalizing or noetherian) if there are no infinite reduction sequences $t_1 \to_{\mathcal{R}} t_2 \to_{\mathcal{R}} \ldots$.

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 [15].

2.1 Basic Narrowing

Basic narrowing is the restriction of narrowing introduced by Hullot [15] which is essentially based on forbidding narrowing steps on terms brought in by instantiation. We use the definition of basic narrowing given in [14], 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 $\mathcal{R}$, $l \to r \ll \mathcal{R}$ denotes that $l \to r$ is a *fresh* variant of a rule in $\mathcal{R}$, i.e., all the variables are *fresh*.

Definition 1 (Basic narrowing). *[14] 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 $\langle s, \sigma \rangle \overset{b}{\leadsto}_{p,\mathcal{R},\theta} \langle t, \sigma' \rangle$ if there exist $p \in \mathcal{P}os_{\Sigma}(s)$, $l \to r \ll \mathcal{R}$, and substitution θ such that $\theta = mgu(s|_p\sigma, l)$, $t = (s[r]_p)$, and $\sigma' = \sigma\theta$.*

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

We say that $\mathcal{R}$ is $(\overset{b}{\leadsto})$-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). [15,2] *Let $\mathcal{R}$ be a TRS. If for every $l \to r \in \mathcal{R}$, all basic narrowing derivations issuing from r terminate, then $\mathcal{R}$ is $(\overset{b}{\leadsto})$-terminating.*

In the literature, this condition has been approximated by requiring that every rhs of a rewrite rule is a variable, in [15], or a constructor term, in [21]. 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 [23]. 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 [23].

Definition 2 (Antecedent of a position). [23] *Let $t \xrightarrow{p}_{l\to r} t'$ be a rewriting step, $v \in \mathcal{P}os(t)$, and $v' \in \mathcal{P}os(t')$. 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 $u' \in \mathcal{P}os_x(r)$ of a variable x in r s.t. $v' = p.u'.w$ and $v = p.u.w$, where $u \in \mathcal{P}os_x(l)$ 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 $u' \in \mathcal{P}os_\Sigma(r)$, 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). [23] *Let $t \overset{b}{\leadsto}^{*}_{\mathcal{R},\sigma} t'$, $v \in \mathcal{P}os(t)$, and $v' \in \mathcal{P}os(t')$. We say v is an antecedent of v' iff v is an antecedent of v' in the rewrite sequence $t\sigma \to^*_{\mathcal{R}} t'$.*

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

$$s \overset{b}{\leadsto}_{p,g\to d,\sigma} t \overset{b}{\leadsto}_{q,l\to r,\theta} u \tag{1}$$

and we are interested in the conditions that allow us to commute the first two steps by first applying to s the rule $l \to r$ and then the rule $g \to d$ 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 $l \to r$ to all the antecedents of q, and then applying $g \to d$ to the resulting term. Let us give an example for motivation.

Example 1. [23] Let us consider the following TRS $\mathcal{R}_4$:

$$\mathcal{R}_4 = \{\ \mathtt{f}(x,x) \ \to\ x\ (\mathrm{r1}) \qquad \mathtt{g}(x,\mathtt{h}(x)) \ \to\ x\ (\mathrm{r2})\ \}$$

and the basic narrowing derivation:

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

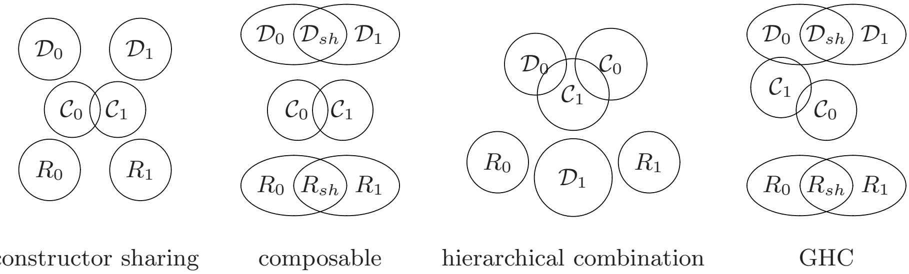

Fig. 1. Standard modular combinations

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

$$\langle \mathtt{h}(\mathtt{f}(\mathtt{0}, x), \mathtt{g}(x, y)), \{\}\rangle \overset{b}{\leadsto}_{q'=2, r2, \{y/\mathtt{h}(x)\}} \langle \mathtt{h}(\mathtt{f}(\mathtt{0}, x), x), \{\}\rangle \overset{b}{\leadsto}_{p=1, r1, \{x/\mathtt{0}\}} \langle \mathtt{h}(x, x), \{x/\mathtt{0}\}\rangle$$

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* $t \overset{b}{\leadsto}_{p, l\to r, \sigma} t' \overset{b}{\leadsto}_{q', g\to d, \theta} u$, *if* $q \in \mathcal{P}os(t)$ *is an antecedent of* q', *then* q *and* p *are necessarily disjoint,* q' *is in the skeleton, and* $q = q'$.

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*

$$\langle s, \theta\rangle \overset{b}{\leadsto}_{p, g\to d, \sigma_1} \langle s[d]_p, \theta\sigma_1\rangle \overset{b}{\leadsto}_{q, l\to r, \sigma_2} \langle s[d]_p[r]_q, \theta\sigma_1\sigma_2\rangle \qquad (2)$$

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:*

$$\langle s, \theta\rangle \overset{b}{\leadsto}_{q, l\to r, \sigma_3} \langle s[r]_q, \theta\sigma_3\rangle \overset{b}{\leadsto}_{p, g\to d, \sigma_4} \langle s[r]_q[d]_p, \theta\sigma_3\sigma_4\rangle \qquad (3)$$

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

4 Modular Termination of Basic Narrowing

Let us recall some standard notions regarding modularity of rewriting, as defined in [20], that will be used throughout the paper. Figure 1 shows diagramatic 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, $\Sigma_0 \cap \Sigma_1 = \varnothing$. 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., $\mathcal{D}_0 \cap \mathcal{D}_1 = \varnothing$.

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 $\mathcal{D}_0 \cap \mathcal{C}_1 = \mathcal{D}_1 \cap \mathcal{C}_0 = \varnothing$ 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 $\mathcal{R}_{sh} = \{l \to r \in \mathcal{R}_0 \cup \mathcal{R}_1 \mid root(l) \in \mathcal{D}_{sh}\}$.

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 $\mathcal{D}_0 \cap \mathcal{D}_1 = \varnothing$ and $\mathcal{C}_0 \cap \mathcal{D}_1 = \varnothing$.

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 $\mathcal{D}_0 \cap \mathcal{D}_1 = \varnothing$, $\mathcal{C}_0 \cap \mathcal{D}_1 = \varnothing$, $R_{sh} = R_0 \cap R_1$ where $R_{sh} = \{l \to r \in \mathcal{R}_0 \cup \mathcal{R}_1 \mid root(l) \in \mathcal{D}_{sh}\}$, 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 [22], 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 [20] illustrates these notions.

Example 2. Consider the following TRSs:

$$\mathcal{R}_+ = \left\{ \begin{array}{c} \mathtt{0} + y \to y \\ \mathtt{s}(x) + y \to \mathtt{s}(x + y) \end{array} \right. \qquad \mathcal{R}_- = \left\{ \begin{array}{c} \mathtt{0} - \mathtt{s}(y) \to \mathtt{0} \\ x - \mathtt{0} \to x \\ \mathtt{s}(x) - \mathtt{s}(y) \to x - y \end{array} \right.$$

$$\mathcal{R}_* = \left\{ \begin{array}{c} \mathtt{0} * y \to \mathtt{0} \\ \mathtt{s}(x) * y \to (x * y) + y \end{array} \right. \qquad \mathcal{R}_{pow} = \left\{ \begin{array}{c} \mathtt{pow}(x, \mathtt{0}) \to \mathtt{s}(\mathtt{0}) \\ \mathtt{pow}(x, \mathtt{s}(y)) \to x * \mathtt{pow}(x, y) \end{array} \right.$$

$$\mathcal{R}_{app} = \left\{ \begin{array}{c} \mathtt{nil} \mathtt{++} ys \to ys \\ (x : xs) \mathtt{++} ys \to x : (xs \mathtt{++} ys) \end{array} \right.$$

$\mathcal{R}_+$ 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 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.*

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

Example 3 (Toyama). Let us consider Toyama's example [25]:

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

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 Toyama's example is not $(\rightarrow)$-terminating. However, it is innermost terminating. This shows that $(\overset{b}{\leadsto})$-termination does not entail $(\rightarrow)$-termination, which suggests that the modularity requirements for $(\overset{b}{\leadsto})$-termination are less restrictive than those of $(\rightarrow)$-termination. Actually, the modularity properties of $(\overset{b}{\leadsto})$-termination are comparable to those of innermost $(\rightarrow)$-termination (see e.g. [20]). The next theorem extends the modularity of $(\overset{b}{\leadsto})$-termination to composable systems.

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

In Section 4.3 we further extend this result up to generalized proper extensions (GPE) [22], a fairly general restriction of HCs. To achieve this result, we proceed as follows. First we prove in Section 4.2 the modularity for generalized *nice* extensions (GNE) [22], a restriction of GPEs. Then, we apply a result from [22] that relates GPEs to GNEs, which delivers the desired result.

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

Definition 4 (Dependency Relation $\trianglerighteq_{\mathcal{R}}$). [22] *For a TRS $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ the dependency relation $\trianglerighteq_{\mathcal{R}}$ is the smallest preorder satisfying the condition $f \trianglerighteq_{\mathcal{R}} g$ whenever there is a rewrite rule $f(s_1, \ldots, s_m) \rightarrow r \in \mathcal{R}$ and $g(t_1, \ldots, t_n)$ is a subterm of r, with $g \in \mathcal{D}$.*

We often omit $\mathcal{R}$ from $\trianglerighteq_{\mathcal{R}}$ 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 $f \trianglerighteq g$. Intuitively, $f \trianglerighteq g$ if the evaluation of f involves a call to g after one or more rewrite steps.

4.2 Nice Extensions

Nice extensions (NE) are a restriction of PEs introduced by Krishna Rao [22]. 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 4.3.

Definition 5 (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., $\mathcal{D}_1^0 = \{f \in \mathcal{D}_1 \mid \exists g \in \mathcal{D}_0, f \trianglerighteq_{\mathcal{R}} g\}$ 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 $\mathcal{R}_1^0 = \{l \to r \in \mathcal{R}_1 \mid root(l) \in \mathcal{D}_1^0\}$ and $\mathcal{R}_1^1 = \{l \to r \in \mathcal{R}_1 \mid root(l) \in \mathcal{D}_1^1\}$.*

Definition 6 (Generalized Nice Extension). *[22] 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 $l \to r \in \mathcal{R}_1$, and for every subterm s of r such that $root(s) \in \mathcal{D}_1^0$, s contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

Figure 4 shows a diagramatic rendition of NEs, and an example can be found in Example 4 later.

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 $\Box$. If C is a context and $\overline{t}$ a list of terms, $C[\overline{t}]$ denotes the result of replacing the holes in C by the terms in $\overline{t}$.

Definition 7 ($\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terms). *Let $(\mathcal{D} \uplus \mathcal{C}, \mathcal{R})$ be the union of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and a GNE $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets $\mathcal{D}_1^0$, $\mathcal{D}_1^1$, $\mathcal{R}_1^0$ and $\mathcal{R}_1^1$ as in Definition 5. 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 $C[s_1, \ldots, s_n]$, where $C \in \mathcal{CC}_{01}$ and the following conditions hold:*

1. *for all $i \in \{1 \cdots n\}$, $root(s_i) \in \mathcal{D}_1^0$, and*
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 $\overset{b}{\leadsto}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ if $\mathcal{R}_1$ is a GNE of $\mathcal{R}_0$.

The reader can check that the right hand sides of the rules in a GNE fulfill the conditions above. In order to prove the $(\overset{b}{\leadsto})$-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 in Section 5 the more general result that derivations starting from $\mathcal{S}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ terms are finite.

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.*

4.3 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 8 (Generalized Proper Extension). *[22] Let $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ be the GHC of a base system $(\mathcal{D}_0 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_0, \mathcal{R}_0)$ and an extension $(\mathcal{D}_1 \uplus \mathcal{D}_{sh} \uplus \mathcal{C}_1, \mathcal{R}_1)$. Define the sets $\mathcal{D}_1^0$, $\mathcal{D}_1^1$, $\mathcal{R}_1^0$ and $\mathcal{R}_1^1$ as in Definition 5. $\mathcal{R}_1$ is a generalized* proper *extension (GPE) of $\mathcal{R}_0$ if each rewrite rule $l \to r \in \mathcal{R}_1^0$ satisfies that, for every subterm t of r such that $root(t) \in \mathcal{D}_1^0$ and $root(t) \trianglerighteq_{\mathcal{R}} root(l)$, t contains no function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

Figure 4 shows a diagramatic rendition of GPEs.

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

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

$\mathcal{R}_!$ 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:

$$\mathcal{R}_1 : \{\mathtt{f(a)} \to \mathtt{f(b)}\} \qquad\qquad \mathcal{R}_0 : \{\mathtt{b} \to \mathtt{a}\}$$

There exists the following infinite basic narrowing derivation

$$\mathtt{f(a)} \overset{b}{\leadsto} \mathtt{f(b)} \overset{b}{\leadsto} \mathtt{f(a)} \overset{b}{\leadsto} \cdots$$

produced by the nesting of a redex w.r.t. $\mathcal{R}_0$ inside the recursive call to `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

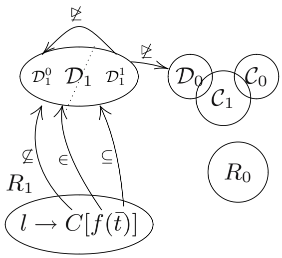

Fig. 2. Nice extension

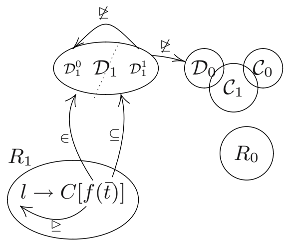

Fig. 3. Proper extension

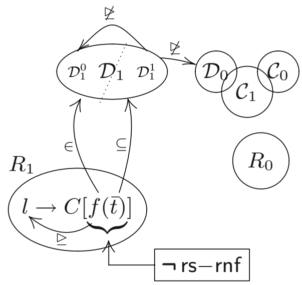

Fig. 4. Relaxed Proper extension

TRS in [26]. 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 $\trianglerighteq$ on defined function symbols, and the rules corresponding to the symbols of every strongly connected component become a module (i.e., a GNE).

Lemma 2. *[22] Let $\mathcal{R}_1$ be a finite TRS such that it is a GPE of $\mathcal{R}_0$. Then $\mathcal{R}_1$ 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 following section, 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 encoding[3] 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.

$$\mathcal{R}_1 : \mathtt{exp}(\mathtt{exp}(\mathtt{g}, X), Y) \to \mathtt{exp}(\mathtt{g}, X{*}Y)$$
$$\mathcal{R}_0 : X{*}X^{-1} \to 1 \qquad X{*}1 \to X \qquad 1{*}X \to X$$

[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.

Basic narrowing trivially terminates on each system separately, since every rhs is clearly unnarrowable. However, their combination $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ is not a PE, since the base defined symbol `*` appears below the extension defined symbol `exp` in a recursive call. It is easy to see that basic narrowing indeed terminates in $\mathcal{R}$, because the outer function symbol `exp` in the recursive invocation occurring in the right hand side of the rule of $\mathcal{R}_1$ is blocked forever. 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 abuse of notation, we apply this notion, with no change, to basic narrowing.

Definition 9 (Root-Stable Rigid Normal Form). [2] *A term s is a* root-stable rigid normal form *($\mathsf{rs{-}rnf}$) 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. $s\theta \stackrel{>\epsilon *}{\to}_{\mathcal{R}} s' \stackrel{b}{\leadsto}_{\epsilon,\mathcal{R},\theta'} s''$.*

Definition 10 (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 5. $\mathcal{R}_1$ 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) $root(t) \in \mathcal{D}_1^0$, (b) t is not a $\mathsf{rs{-}rnf}$, and (c) $root(t) \trianglerighteq_{\mathcal{R}} root(l)$, t does not contain a function symbol of $\mathcal{D}_0 \cup \mathcal{D}_1^0$ strictly below its root.*

Figure 4 shows a diagramatic rendition of GRPEs, and the reader can check that the TRS of Example 6 is indeed a GRPE. In the following, we show that ($\stackrel{b}{\leadsto}$)-termination is modular for RPEs by showing first its modularity for GRNEs, and then establishing a relation between GRPEs an GRNEs. The reasoning is similar to the one followed in Section 4.3.

Definition 11 (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 an 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 5. $\mathcal{R}_1$ is a generalized relaxed* nice *extension (GRNE) of $\mathcal{R}_0$ if it is a GRPE, and for every rewrite rule $l \to r \in \mathcal{R}_1$ the following condition holds:*

(N1) *for each subterm t of r such that t is not a $\mathsf{rs{-}rnf}$ and $root(t) \in \mathcal{D}_1^0$, 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 12 ($\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ 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 an 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 5. We define $\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ as the set of all terms of the form $C[s_1, \ldots, s_n]$, 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* $i \in \{1 \cdots n\}$, $root(s_i) \in \mathcal{D}_1^0$, s_i *is not a* $\mathsf{rs{-}rnf}$, *and*
3. s_i *contains no function symbol of* $\mathcal{D}_0 \cup \mathcal{D}_1^0$ *strictly below its root.*

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

Lemma 3. *If* $t \in \mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ *and* $t \overset{b}{\leadsto}_{\mathcal{R}_0 \cup \mathcal{R}_1, \theta} t'$, *then* $t' \in \mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$.

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

Definition 13 (Abstract Reduction System). *An abstract reduction system (ARS) is a structure* $\mathcal{A} = (A, \{\to_\alpha \mid \alpha \in I\})$ *consisting of a set* A *and a set of binary relations* $\to_\alpha$ *on* A, *indexed by a set* I. *We write* $(A, \to_1, \to_2)$ *instead of* $(A, \{\to_\alpha \mid \alpha \in \{1, 2\}\})$.

Definition 14 (Quasi-commutation). *[6] Let* $\to_0$ *and* $\to_1$ *be two relations on a set* S. *The relation* $\to_1$ *quasi-commutes over* $\to_0$ *if, for all* $s, u, t \in S$ *s.t.* $s \to_0 u \to_1 t$, *there exists* $v \in S$ *s.t.* $s \to_1 v \to_{01}^* t$, *where* $\to_{01}^*$ *is the transitive-reflexive closure of* $\to_0 \cup \to_1$.

Theorem 4. *[6] If the relations* $\to_0$ *and* $\to_1$ *in the ARS(*S, $\to_0$, $\to_1$*) are strongly normalizing and* $\to_1$ *quasi-commutes over* $\to_0$, *the relation* $\to_0 \cup \to_1$ *is strongly normalizing too.*

We now define an ARS with skeleton–environment pairs as elements, where the skeletons come from the set $\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ of terms, and the relationships $\to_0$ and $\to_1$ of the ARS are restrictions of basic narrowing.

Definition 15. *Let* $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$ *be a generalized relaxed nice combination. We define the ARS* $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1) = (\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1} \times Subst, \to_0, \to_1)$, *where the relations* $\to_0$ *and* $\to_1$ *are defined as follows. Let* $s = C[s_0, \ldots, s_n]$ *be a term in* $\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$. *Then*

1. $\langle C[s_0, \ldots, s_n], \sigma \rangle \to_0 \langle C'[s_0, \ldots, s_n], \theta\sigma \rangle$ *if* $\langle C[s_0, \ldots, s_n], \sigma \rangle \overset{b}{\leadsto}_{\mathcal{R}_0 \cup \mathcal{R}_1^1, \theta} \langle C'[s_0, \ldots, s_n], \sigma\theta \rangle$ *is a basic narrowing step given within the context* C
2. $\langle C[s_0, \ldots, s_n], \sigma \rangle \to_1 \langle C[s_0, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n], \theta\sigma \rangle$ *if* $\langle C[s_0, \ldots, s_n], \sigma \rangle \overset{b}{\leadsto}_{\mathcal{R}_1, \theta} \langle C[s_0, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n], \theta\sigma \rangle$ *is a basic narrowing step given at a subterm* s_i, *with* $i \in \{0, \ldots, n\}$.

The relation $\to_1 \cup \to_0$ is exactly the basic narrowing relation over $\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$. In the following we establish that both $\to_0$ and $\to_1$ are terminating relations.

Lemma 4. *Given the ARS* $\mathcal{A}(\mathcal{R}_0, \mathcal{R}_1)$ *of Definition 15, the relations* $\to_0$ *and* $\to_1$ *are terminating if* $\mathcal{R}_0$ *and* $\mathcal{R}_1$ *are* $(\overset{b}{\leadsto})$*-terminating.*

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 5. *Given the ARS $\mathcal{A}(\mathcal{R}_0,\mathcal{R}_1)$ of Definition 15, the relation $\rightarrow_1$ quasi-commutes over the relation $\rightarrow_0$.*

Then, as a straightforward consequence of Theorem 4 and Theorem 5, 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 $\mathcal{S}^{\mathsf{rs-rnf}}_{\mathcal{R}_0 \cup \mathcal{R}_1}$ 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 [22] extending them to our generalized relaxed nice extensions.

Lemma 5. *Let $\mathcal{R}_1$ be a finite TRS such that it is a GRPE of $\mathcal{R}_0$. $\mathcal{R}_1$ can be seen as a finite pyramid of GRNEs.*

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

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 [15,23,19]. 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 [15,2]. Narrowing has received much attention due to the different applications, such as automated proofs of termination [5], execution of multiparadigm programming languages [13,17], symbolic reachability [18], verification of cryptographic protocols [10], equational unification [15], equational constraint solving [3,4], and model checking [11], among others. Termination of narrowing is, therefore, of much interest to these applications.

References

1. Alpuente, M., Escobar, S., Iborra, J.: Modular termination of basic narrowing. Technical Report DSIC-II/04/08, Universidad Politécnica de Valencia (2007)
2. Alpuente, M., Escobar, S., Iborra, J.: Termination of Narrowing revisited. Theoretical Computer Science (to appear, 2008)
3. Alpuente, M., Falaschi, M., Gabbrielli, M., Levi, G.: The semantics of equational logic programming as an instance of CLP. In: Logic Programming Languages, pp. 49–81. The MIT Press, Cambridge (1993)
4. Alpuente, M., Falaschi, M., Levi, G.: Incremental Constraint Satisfaction for Equational Logic Programming. Theoretical Computer Science 142(1), 27–57 (1995)
5. Arts, T., Giesl, J.: Termination of term rewriting using dependency pairs. Theor. Comput. Sci. 236(1-2), 133–178 (2000)
6. Bachmair, L., Dershowitz, N.: Commutation, transformation, and termination. In: Proc. of the 8th Int'l Conf. on Automated Deduction, January 1986, pp. 5–20 (1986)
7. Comon-Lundh, H.: Intruder Theories (Ongoing Work). In: Walukiewicz, I. (ed.) FOSSACS 2004. LNCS, vol. 2987, pp. 1–4. Springer, Heidelberg (2004)
8. Cortier, V., Delaune, S., Lafourcade, P.: A Survey of Algebraic Properties used in Cryptographic Protocols. Journal of Computer Security 14(1), 1–43 (2006)
9. Dershowitz, N., Jouannaud, J.-P.: Rewrite systems. In: Handbook of Theoretical Computer Science, vol. B, pp. 244–320. Elsevier Science, Amsterdam (1990)
10. Escobar, S., Meadows, C., Meseguer, J.: A Rewriting-Based Inference System for the NRL Protocol Analyzer and its Meta-Logical Properties. TCS 367 (2006)
11. Escobar, S., Meseguer, J.: Symbolic model checking of infinite-state systems using narrowing. In: Baader, F. (ed.) RTA 2007. LNCS, vol. 4533, pp. 153–168. Springer, Heidelberg (2007)
12. Fay, M.: First-Order Unification in an Equational Theory. In: Fourth Int'l Conf. on Automated Deduction, pp. 161–167 (1979)
13. Hanus, M.: The Integration of Functions into Logic Programming: From Theory to Practice. Journal of Logic Programming 19&20, 583–628 (1994)
14. Hölldobler, S. (ed.): Foundations of Equational Logic Programming. LNCS, vol. 353. Springer, Heidelberg (1989)
15. Hullot, J.-M.: Canonical Forms and Unification. In: Bibel, W. (ed.) CADE 1980. LNCS, vol. 87, pp. 318–334. Springer, Heidelberg (1980)
16. Hullot, J.-M.: Compilation de Formes Canoniques dans les Théories ǵquationelles. Thése de Doctorat de Troisième Cycle. PhD thesis, Université de Paris Sud, Orsay (France) (1981)
17. Meseguer, J.: Multiparadigm logic programming. In: Kirchner, H., Levi, G. (eds.) ALP 1992. LNCS, vol. 632, pp. 158–200. Springer, Heidelberg (1992)
18. Meseguer, J., Thati, P.: Symbolic reachability analysis using narrowing and its application to verification of cryptographic protocols. HOSC, 123–160 (2007)
19. Middeldorp, A., Hamoen, E.: Completeness Results for Basic Narrowing. J. of Applicable Algebra in Engineering, Comm. and Computing 5, 313–353 (1994)
20. Ohlebusch, E.: Advanced Topics in Term Rewriting. Springer, Heidelberg (2002)
21. Prehofer, C.: On Modularity in Term Rewriting and Narrowing. In: Jouannaud, J.-P. (ed.) CCL 1994. LNCS, vol. 845, pp. 253–268. Springer, Heidelberg (1994)
22. Krishna Rao, M.R.K.: Modular proofs for completeness of hierarchical term rewriting systems. Theoretical Computer Science (January 1995)

23. Réty, P.: Improving Basic Narrowing Techniques. In: Lescanne, P. (ed.) RTA 1987. LNCS, vol. 256. Springer, Heidelberg (1987)
24. TeReSe (ed.): Term Rewriting Systems. Cambridge University Press, Cambridge (2003)
25. Toyama, Y.: Counterexamples to termination for the direct sum of term rewriting systems. Inf. Process. Lett. 25(3), 141–143 (1987)
26. Urbain, X.: Modular & incremental automated termination proofs. Int. J. Approx. Reasoning 32(4), 315–355 (2004)