Generalized innermost rewriting

Jaco van de Pol[1,2] and Hans Zantema[2]

[1] Department of Software Engineering, CWI, P.O. Box 94.079, 1090 GB Amsterdam, The Netherlands, Jaco.van.de.Pol@cwi.nl

[2] Department of Computer Science, TU Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, H.Zantema@tue.nl

Abstract. We propose two generalizations of innermost rewriting for which we prove that termination of innermost rewriting is equivalent to termination of generalized innermost rewriting. As a consequence, by rewriting in an arbitrary TRS certain non-innermost steps may be allowed by which the termination behavior and efficiency is often much better, but never worse than by only doing innermost rewriting.

1 Introduction

In term rewriting one can rewrite according various reduction strategies, for instance innermost or outermost. It may occur that by one strategy a normal form is reached while rewriting according another strategy may go on forever. For instance, innermost rewriting of $f(a)$ by the TRS consisting of the two rules $a \to b, f(a) \to f(a)$ yields a normal form in one step while outermost rewriting goes on forever. By the TRS consisting of the two rules $a \to a, f(x) \to b$ the behavior of $f(a)$ is opposite: now innermost rewriting goes on forever while outermost rewriting yields a normal form in one step. We say that a particular strategy is not worse than another strategy if for every term for which the latter yields a normal form, the same holds for the former. The above examples show that innermost and outermost are incomparable. For orthogonal TRSs it is known that no strategy is worse than innermost ([8]); later it was proved that the same result holds more generally for non-overlapping TRSs ([5]). In practice, many implementations use innermost rewriting since it is easy to implement and often efficient. On the other hand, for implementing lazy evaluation for functional programming it is essential to do non-innermost rewriting since otherwise computations will not terminate. The main idea of lazy rewriting ([3]) is that doing a computation is postponed until the result is required for continuation.

In this paper we consider strategies for arbitrary TRSs allowing overlaps and even non-confluence, as often occurs in applications in theorem proving. For instance, the natural rules to obtain conjunctive normal forms are not confluent. We present two generalizations of innermost rewriting that are always provably not worse than innermost, and allow non-innermost steps as they are preferred in lazy rewriting. In the usual definition of innermost rewriting it is required that all proper subterms of a redex are in normal form; in our definition of generalized innermost rewriting we require this only for particular proper subterms.

As an example, consider the following TRS for computation of factorials:

$$\begin{aligned} \mathsf{fac}(x) &\to \mathsf{if}(\mathsf{eq}(x,0), \mathsf{succ}(0), x * \mathsf{fac}(\mathsf{pred}(x))) \\ \mathsf{if}(\mathsf{true}, x, y) &\to x \\ \mathsf{if}(\mathsf{false}, x, y) &\to y \end{aligned}$$

completed by standard rules for eq, pred and $*$. Now $\mathsf{fac}(\mathsf{succ}^n(0))$ is not weakly innermost normalizing, for any $n \geq 0$, but it is weakly generalized innermost normalizing. Moreover, straightforward implementations of generalized innermost rewriting easily find the corresponding normal form.

In case of constructor systems that are either right-linear or non-root-overlapping, our generalization corresponds to arbitrary rewriting. But our results also apply for TRSs not being constructor systems or having overlaps. For instance, in the above example we may have rules like $(x * y) * z \to x * (y * z)$.

Apart from only considering whether a reduction will terminate or not, also lengths of reductions to normal form may be considered. Then it is natural for calling one strategy not worse than another strategy to require additionally that if both strategies yield a normal form then the former strategy does not take more steps. For this extra requirement it is obvious that duplicating rules should be avoided. By doing so we prove that indeed this stronger result of being not worse than innermost is obtained for generalized innermost rewriting.

Many implementations of rewriting including OBJ ([4, 7]) make use of similar modifications of the innermost strategy to achieve better termination behavior or efficiency. The basis of our work was in [10, 11], where JITty strategy annotations and a corresponding implementation was proposed to postpone computation steps in a slightly more general way than eager annotations in OBJ. All these implementations are essentially deterministic and apply particular cases of generalized innermost rewriting, while often non-innermost steps are done. Typically, the termination behavior and efficiency of these implementations is much better, but by our results never worse than by only doing innermost rewriting.

The outline of the paper is as follows. After recalling some standard notation in Section 2, we introduce our basic generalization of innermost rewriting in Section 3. Unfortunately, some further restrictions have to be given in order to have the desired properties, as we show by an example. This is done in two ways. In Section 4 this is done by avoiding duplication, by which also results are obtained involving the number of rewrite steps. In Section 5 this is done by avoiding root overlaps and applying priority of rules. In particular an open problem from [10] is solved.

2 Basic notation

As usual, for binary relations $\to, \to'$ on a set T we define relation composition by $\to \cdot \to' := \{(t, t') \mid \exists t'' : t \to t'' \wedge t'' \to' t' \}$. With $\to^n$ we denote the n-fold relation composition, and $\to^+$ and $\to^*$ denote the transitive, and the transitive reflexive closure of $\to$, respectively. Finally, $\leftarrow$ denotes the inverse of

$\to$. Using these notations many properties can be expressed shortly, for instance, local confluence of $\to$ is expressed by $\leftarrow \cdot \to \subseteq \to^* \cdot \leftarrow^*$.

An object t is in *normal form* in $\to$, if no s exists satisfying $t \to s$. The set $\mathsf{NF}(\to)$ denotes the set of all $\to$-normal forms. An object $t \in T$ is *weakly normalizing* in $\to$, denoted by $\mathsf{WN}(t, \to)$, if there exists a reduction $t \to^* s$ and $s \in \mathsf{NF}(\to)$. An object $t \in T$ is *terminating* in $\to$, denoted by $\mathsf{SN}(t, \to)$, if there is no infinite $\to$-reduction sequence starting in t.

In the sequel, we consider a fixed set of function symbols and a fixed set of variables $\mathcal{V}$, from which terms are built as usual. By $\mathsf{arity}(f)$ we denote the number of arguments expected by f. The set of variables occurring in a term t is denoted by $\mathsf{Var}(t)$; we will use $\mathsf{LinVar}(t)$ to denote the set of variables that occur exactly once in t. A substitution is a mapping from variables to terms, and application of substitution σ to a term t is denoted by $t\sigma$, or by t^σ when σ is complex expression.

A *position* is defined to be a list of positive integers, as in [1]; ε denotes the empty list, corresponding to the root position. The set of valid positions in a term t is denoted by $\mathsf{Pos}(t)$; the subterm at position p in a term t is denoted by $t|_p$; its root symbol is denoted by $(t)_p$. Replacing the subterm at position p in C by s is denoted by $C[s]_p$, or simply $C[s]$, where $C[]$ is called the context. Two positions that are not comparable in the prefix order are called *disjoint*.

We consider rewriting w.r.t. a fixed TRS, consisting of a set of rules of the form $\ell \to r$, such that $\ell \notin \mathcal{V}$ and $\mathsf{Var}(r) \subseteq \mathsf{Var}(\ell)$. Rules $\ell_1 \to r_1$ and $\ell_2 \to r_2$ are *root overlapping* if for some σ and τ, $\ell_1\sigma = \ell_2\tau$.

Let Def denote the set of *defined symbols*, that is, symbols that occur as the root symbol of the left hand side of a rule. A TRS is called a *constructor system* if for all left hand sides ℓ the root symbol is the only occurrence of a symbol in Def.

3 Generalized innermost rewriting

Recall that we consider a fixed TRS.

Definition 1. *A term t rewrites to a term u, written $t \to u$, if*

- $t = C[\ell\sigma]$ *and* $u = C[r\sigma]$ *for a rule* $\ell \to r$*, context* C *and substitution* σ*.*

The subterm $\ell\sigma$ of t is called the corresponding redex*. In the sequel, the set* NF *always denotes the set of normal forms* $\mathsf{NF}(\to)$*.*

Definition 2. *A term t rewrites* innermost *to a term u, notation $t \to_i u$, if*

- $t = C[\ell\sigma]$ *and* $u = C[r\sigma]$ *for a rule* $\ell \to r$*, context* C *and substitution* σ*; and*
- $\ell\sigma|_p \in \mathsf{NF}$ *for all* $p \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$*.*

It is easy to see that this corresponds to the usual notion of innermost rewriting. It is also easy to see that $\mathsf{NF}(\to_i) = \mathsf{NF}(\to)$. We now introduce a generalization of innermost rewriting in which the second condition is slightly weakened.

Definition 3. *A term t rewrites* generalized innermost *to a term u, notation $t \rightarrow_g u$, if*

- *$t = C[\ell\sigma]$ and $u = C[r\sigma]$ for a rule $\ell \rightarrow r$, context C and substitution σ; and*
- *$\ell\sigma|_p \in \mathsf{NF}$, for all $p \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$, such that $(\ell)_p \in \mathsf{Def}$.*

Clearly, $\rightarrow_i \subseteq \rightarrow_g \subseteq \rightarrow$, hence $\mathsf{NF}(\rightarrow_g) = \mathsf{NF}(\rightarrow)$. Note that for constructor systems, $\rightarrow_g = \rightarrow$, i.e. generalized innermost rewriting corresponds to general rewriting.

We hope that generalized innermost rewriting has better normalization properties than innermost rewriting. This means first of all that if a term t is $\rightarrow_i$-terminating, then it should be $\rightarrow_g$-terminating. However, this is not always the case, as is witnessed by the following example inspired by [9]:

Example 4. Consider the TRS consisting of the three rules

$$f(a, b, x) \rightarrow f(x, x, x), \quad c \rightarrow a, \quad c \rightarrow b$$

$\rightarrow_i$ is terminating on $f(a, b, c)$, while $\rightarrow_g$ is not terminating on $f(a, b, c)$.

We see in this example that redexes can be copied by the first rule, and that the copies can behave differently, because the last rules are root-overlapping. Both ingredients are essential for this counter example. In Section 4, we restrict the generalized innermost strategy to avoid duplication of redexes. In Section 5 we restrict the generalized innermost strategy to avoid root-overlaps by assuming a priority on the rules. In both cases we prove that innermost termination coincides with termination of the restricted generalized innermost reduction relation. The proofs of these results are quite different, and the sections can be read independently.

Moreover, if redex duplication is avoided, the number of generalized innermost steps is bounded by the number of innermost steps. In case the TRS is non-root-overlapping, we prove equivalence of $\rightarrow_i$-termination and $\rightarrow_g$-termination. The latter section also solves an open problem in [10].

4 Avoiding duplication

In case one wants short reductions to normal form it is clear that duplicating rules should be avoided. For instance, in rewriting the term $f(a)$ by the TRS consisting of the two rules $f(x) \rightarrow g(x, x), a \rightarrow b$ we prefer innermost reduction since otherwise a will be duplicated before it is rewritten, requiring one more rewrite step afterward. Therefore we now adjust our definition of generalized innermost rewriting to a non-duplicating variant as follows.

Definition 5. *A term t rewrites non-dup-generalized innermost to a term u, notation $t \rightarrow_{ndg} u$, if*

- *$t = C[\ell\sigma]$ and $u = C[r\sigma]$ for a rule $\ell \rightarrow r$, context C and substitution σ; and*

– $\ell\sigma|_p \in \mathsf{NF}$ *for all* $p \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$ *for which either*
  - $(\ell)_p \in \mathsf{Def}$*, or*
  - $(\ell)_p$ *is a variable occurring more than once in* r*.*

Clearly, $\to_i \subseteq \to_{ndg} \subseteq \to_g \subseteq \to$, hence $\mathsf{NF}(\to_{ndg}) = \mathsf{NF}(\to)$. Compared to generalized innermost rewriting the extra condition is that subterms on variable positions with multiple occurrences in r should be in normal form. One can argue this is more related to non-right-linearity than to duplication, but since the intension is to avoid duplicating steps we chose to call this 'non-dup'. The desired theorem (Theorem 9) does not hold if we weaken the extra condition to non-duplication in the sense that the variable occurs not more often in r than in ℓ: in the TRS

$$f(x,x) \to g(x,x),\ a \to b,\ a \to c,\ g(b,c) \to g(b,c)$$

the term $f(a,a)$ is innermost terminating while it admits an infinite generalized innermost reduction.

Note that non-dup-generalized innermost rewriting applies to all TRSs, the only extra restriction is that a subterm in a particular position in a redex should be in normal form. For non-dup-generalized innermost rewriting we will show that it is not worse than innermost without any restriction on the order of the rules, and moreover, the lengths of reductions to normal form by this strategy are not worse than by the innermost strategy. First we need some lemmas.

Lemma 6. *Let* t' *be an innermost redex of* $t = C[t']$*, and* $t \to_i^n u$ *for some term* u *and* $n \geq 0$*. Then either*

– $u = C'[t']$ *for some context* C' *and* $C[t''] \to_i^n C'[t'']$ *for any term* t''*, or*
– *there exists* v *such that* $t' \to_i v$ *and* $C[t'] \to_i C[v] \to_i^{n-1} u$*.*

Proof. Induction on n. For $n = 0$ we choose $C' = C$ and we are in the first case. If $n > 0$ and t' is rewritten in the first step then we are in the second case. In the remaining case another redex is rewritten in the first step of $t \to_i^n u$, say t_1 rewriting to t_2. Assume t' is left from t_1 in t; if it is right the argument is similar. So we have

$$t \ = \ D[t', t_1] \ \to_i \ D[t', t_2] \ \to_i^{n-1} \ u.$$

We apply the induction hypothesis to $D[t', t_2] \to_i^{n-1} u$. If the first case holds then we obtain $u = C'[t']$ and $D[t'', t_2] \to_i^{n-1} C'[t'']$, hence $C[t''] = D[t'', t_1] \to_i D[t'', t_2] \to_i^{n-1} C'[t'']$, and we are in the first case. If the second case holds for $D[t', t_2] \to_i^{n-1} u$ then we have v satisfying $t' \to_i v$ and $D[t', t_2] \to_i D[v, t_2] \to_i^{n-2} u$, yielding $C[t'] \ = \ D[t', t_1] \ \to_i \ D[v, t_1] \ \to_i \ D[v, t_2] \ \to_i^{n-2} \ u$ by which we are in the second case, concluding the proof. □

Lemma 7. *Let* $t \to_{ndg} u \to_i^n w$ *for terms* t, u, w *and* $n \geq 0$*, and* $t \not\to_i u$*. Then*

$$t \to_i^+ \cdot \to_{ndg} \cdot \to_i^k \cdot \leftarrow_i^m w$$

for some $k \geq n - 1$ *and* m *being either 0 or 1.*

Proof. Write $t = \overline{C}[\ell\sigma]$ and $u = \overline{C}[r\sigma]$ for a rule $\ell \to r$ and a substitution σ, satisfying $\ell\sigma \to_{ndg} r\sigma$ and $\ell\sigma \not\to_i r\sigma$. Hence $\ell\sigma$ admits a non-root reduction step, say on position $q \neq \epsilon$. If $q \in \mathsf{Pos}(\ell)$ and $\ell|_q$ is not a variable then $\ell|_q \in \mathsf{Def}$ since $\ell\sigma$ admits a reduction step on position q, but then $\ell\sigma|_q \in \mathsf{NF}$ by the definition of $\to_{ndg}$, contradiction. Hence either $\ell|_q$ is a variable or $q \notin \mathsf{Pos}(\ell)$. In the latter case there is a position $p \in \mathsf{Pos}(\ell)$ such that $\ell|_p$ is a variable and $\ell\sigma|_p$ admits a reduction step. So in both cases we have such a variable $x = \ell|_p$ for which $x\sigma \notin \mathsf{NF}$. Since $\ell\sigma \to_{ndg} r\sigma$ and $x\sigma \notin \mathsf{NF}$ we conclude that x occurs at most once in r. Write $\ell = C[x, \ldots, x]$ where x does not occur in C. Choose u' such that $x\sigma \to_i u'$. Define τ by $x\tau = u'$ and $y\tau = y\sigma$ for $y \neq x$. Then

$$t = \overline{C}[\ell\sigma] = \overline{C}[C[x\sigma, \ldots, x\sigma]] \to_i^+ \overline{C}[C[u', \ldots, u']] = \overline{C}[\ell\tau] \to_{ndg} \overline{C}[r\tau].$$

In case x does not occur in r we have $\overline{C}[r\tau] = \overline{C}[r\sigma] = u$ and we are done, choosing $m = 0, k = n$.

In the remaining case x occurs exactly once in r, so $r = D[x]$ where x does not occur in D. Let t' be the redex corresponding to $x\sigma \to_i u'$, so $x\sigma = E[t'] \to_i E[t''] = u'$ for some context E and some term t''. Now we apply Lemma 6 to $u = \overline{C}[r\sigma] = \overline{C}[D\sigma[E[t']]] \to_i^n w$, yielding two cases.

In the first case we obtain C' satisfying $w = C'[t']$ and $D\sigma[E[t'']] \to_i^n C'[t'']$. Now we obtain $k = n$ and $m = 1$ in $w = C'[t'] \to_i C'[t'']$ and

$$\overline{C}[r\tau] = \overline{C}[D\sigma[x\tau]] = \overline{C}[D\sigma[u']] = \overline{C}[D\sigma[E[t'']] \to_i^n C'[t'']$$

and we are done.

In the remaining second case of applying Lemma 6 we obtain a term v satisfying $t' \to_i v$ and $u = \overline{C}[D\sigma[E[t']]] \to_i \overline{C}[D\sigma[E[v]]] \to_i^{n-1} w$. Note that it may be the case that $v \neq t''$. Define ρ by $x\rho = E[v]$ and $y\rho = y\sigma$ for $y \neq x$. Then

$$t = \overline{C}[\ell\sigma] \to_i^+ \overline{C}[\ell\rho] \to_{ndg} \overline{C}[r\rho] = \overline{C}[D\rho[x\rho]] = \overline{C}[D\sigma[E[v]]] \to_i^{n-1} w,$$

by which we are done choosing $m = 0, k = n - 1$. □

Lemma 7 is the key lemma for our Theorems 9 and 12; the rest of the proofs of these theorems and corresponding lemmas hold for arbitrary finitely branching ARSs $\to_i$ and $\to_{ndg}$ satisfying the property of Lemma 7.

Lemma 8. *Let t be a term and $n \geq 0$. Assume $t \to_{ndg}^n u$ for some term u. Then there is a term v satisfying $t \to_i^n v$.*

Proof. Induction on n. For $n = 0$ it is trivial, for $n > 0$ assume $t \to_{ndg} t' \to_{ndg}^{n-1} u$. Applying the induction hypothesis on $t' \to_{ndg}^{n-1} u$ yields a term u' satisfying $t' \to_i^{n-1} u'$. If $t \to_i t'$ we are done; in the remaining case we apply Lemma 7 on $t \to_{ndg} t' \to_i^{n-1} u'$ yielding $t \to_i^+ t'' \to_{ndg} \cdot \to_i^k u''$ for some u'' and $k \geq n - 2$. For $n = 1$ we are done. For $n > 1$ we apply the induction hypothesis on the first $n - 1$ steps of $t'' \to_{ndg} \cdot \to_i^k u''$, note that all $\to_i$ steps are $\to_{ndg}$ steps too. This yields a term v' satisfying $t \to_i^+ t'' \to_i^{n-1} v'$; now we can choose v to be the term obtained after n steps in this reduction. □

Theorem 9. *Let the given TRS be finite, and let t be a term. Then $\rightarrow_i$ is terminating on t if and only if $\rightarrow_{ndg}$ is terminating on t.*

Proof. The 'if'-part trivially holds since every $\rightarrow_i$ step is a $\rightarrow_{ndg}$ step. For the 'only if'-part assume t admits an infinite $\rightarrow_{ndg}$-reduction while $\rightarrow_i$ is terminating on t. So the reduction graph of t with respect to $\rightarrow_i$ does not contain infinite paths. Since the TRS is finite, this graph is finitely branching. Hence by König's lemma this graph is finite and acyclic. Hence a number N exists such that all $\rightarrow_i$ reductions of t have length $\leq N$. Since t admits an infinite $\rightarrow_{ndg}$-reduction there is a term u satisfying $t \rightarrow_{ndg}^{N+1} u$, contradicting Lemma 8. □

Theorem 9 does not hold if $\rightarrow_{ndg}$ is replaced by $\rightarrow_g$, as witnessed by the TRS of Example 4. Hence the restriction in the definition of $\rightarrow_{ndg}$ that $\ell\sigma|_p \in \mathsf{NF}$ if $\ell|_p$ is a variable occurring more than once in r, is essential for Theorem 9.

Next we consider lengths of reductions: we will show that if t reduces by $\rightarrow_{ndg}$ in n steps to a normal form, then t admits either an infinite innermost reduction or an innermost reduction of at least n steps to the same normal form.

Lemma 10. *Let t, w be terms, let $\rightarrow_i$ be terminating on t and let $t \rightarrow_{ndg} \cdot \rightarrow_i^n w$ for $n \geq 0$. Then $t \rightarrow_i^{n'} \cdot \leftarrow_i^* w$ for some $n' > n$.*

Proof. We apply induction on $\rightarrow_i$, i.e., in proving that $t \rightarrow_{ndg} \cdot \rightarrow_i^n w$ implies $t \rightarrow_i^{n'} \cdot \leftarrow_i^* w$ we assume the induction hypothesis that a similar property replacing t by t' holds for all n and all t' satisfying $t \rightarrow_i^+ t'$. So assume $t \rightarrow_{ndg} u \rightarrow_i^n w$. If $t \rightarrow_i u$ we are done, otherwise we apply Lemma 7 yielding t', t'' satisfying $t \rightarrow_i^+ t' \rightarrow_{ndg} \cdot \rightarrow_i^k t'' \leftarrow_i^* w$ for $k \geq n-1$. Now applying the induction hypothesis on $t' \rightarrow_{ndg} \cdot \rightarrow_i^k t''$ yields $t \rightarrow_i^+ t' \rightarrow_i^{k'} \cdot \leftarrow_i^* t'' \leftarrow_i^* w$ for $k' > k$. Since $1 + k' > k + 1 \geq n$ we are done. □

Lemma 11. *Let $\rightarrow_{ndg}$ be terminating on a term t and $t \rightarrow_{ndg}^n v$ for some $n \geq 0$. Then $t \rightarrow_i^k \cdot \leftarrow_i^* v$ for some $k \geq n$.*

Proof. Induction on n. For $n = 0$ it is trivial, for $n > 0$ assume $t \rightarrow_{ndg} u \rightarrow_{ndg}^{n-1} v$. Observe that $\rightarrow_{ndg}$ is terminating on u. Applying the induction hypothesis on $u \rightarrow_{ndg}^{n-1} v$ yields $k' \geq n-1$ and w satisfying $u \rightarrow_i^{k'} w$ and $v \rightarrow_i^* w$. Applying Lemma 10 on $t \rightarrow_{ndg} u \rightarrow_i^{k'} w$ yields $t \rightarrow_i^k \cdot \leftarrow_i^* w$ for $k > k'$. Since $v \rightarrow_i^* w$ and $k \geq k' + 1 \geq n$ we are done. □

Theorem 12. *Let the given TRS be finite, let t be a term and let u be a normal form such that $t \rightarrow_{ndg}^n u$. Then either t admits an infinite $\rightarrow_i$-reduction, or $t \rightarrow_i^k u$ for $k \geq n$.*

Proof. Assume t does not admit an infinite $\rightarrow_i$-reduction. Then $\rightarrow_i$ terminates on t, and by Theorem 9 also $\rightarrow_{ndg}$ terminates on t. Then by Lemma 11 we have $t \rightarrow_i^k \cdot \leftarrow_i^* u$ for some $k \geq n$. Since u is a normal form we have $t \rightarrow_i^k u$. □

Hence a $\to_{ndg}$-reduction to normal form is never worse (counted in number of steps) than an innermost reduction to the same normal form.

Theorem 12 does not hold if $\to_{ndg}$ is replaced by $\to_g$: in the TRS consisting of the two rules

$$f(x) \to g(x,x),\ a \to b,$$

the term $f(a)$ admits a three step $\to_g$-reduction

$$f(a) \to_g g(a,a) \to_g g(b,a) \to_g g(b,b),$$

while the only innermost reduction $f(a) \to_i f(b) \to_i g(b,b)$ of $f(a)$ contains only two steps. Clearly for obtaining short reduction sequences in duplicating positions in left hand sides the argument first should be in normal form, as is required by the restriction in the definition of $\to_{ndg}$ that $\ell\sigma|_p \in \mathsf{NF}$ if $\ell|_p$ is a variable occurring more than once in r.

On can wonder whether the finiteness condition in Theorems 9 and 12 are essential. It is claimed by Vincent van Oostrom that it is not. However, for proving so a different approach will be required: the abstract reduction property described in Lemma 7 is not sufficient to conclude the properties claimed in Theorems 9 and 12 in case of infinite branching. For instance, by defining the abstract reduction systems on natural numbers

$$\to_i = \{(0,n) \mid n > 0\} \cup \{(n+1,n) \mid n > 0\}, \quad \to_{ndg} = \to_i \cup \{(0,0)\}$$

the property described in Lemma 7 holds but the properties claimed in Theorems 9 and 12 do not. This example is due to Vincent van Oostrom.

5 Avoiding root overlaps

In this section we will deal with the case that top rewrite steps are deterministic. This holds if the TRS is non-root-overlapping. For general TRSs this can be forced by fixing a priority on the rules. In this case, we can show that generalized innermost reduction steps commute in a proper way with parallel innermost reduction steps. We will first identify the required commutation diagram in an abstract setting (Lemma 13).

In many cases, the non-root overlapping criterion is too restrictive. In order to apply our theory to any TRS, we will assume that overlapping rules will be applied in a fixed order. For implementations, this is a natural restriction. This idea is implemented by a partial order on the rules, following *priority rewrite systems* [2]. For a fixed TRS, any partial order gives a particular strategy $\to_{g>}$ (Definition 15). Finally, we show that the commutation diagram holds for these strategies (Lemma 20). As a conclusion, we obtain that if innermost rewriting is terminating, then generalized innermost rewriting with ordered rules is terminating, Theorem 21.

The following lemma holds for all ARSs. One can think of $\to$ and $\twoheadrightarrow\!\!\!\!\!\!\shortmid\;$ as innermost rewriting and parallel innermost rewriting, respectively, and $\Rightarrow$ as an extension of it, such as generalized innermost rewriting. Later we will use another instance.

Lemma 13. *Let binary relations $\Rightarrow$, $\rightarrow$ and $\twoheadrightarrow_n$ $(n \geq 0)$ be given. We write $\twoheadrightarrow$ for $\bigcup_{n\geq 0} \twoheadrightarrow_n$. Assume $\rightarrow \subseteq \Rightarrow$, $\rightarrow \subseteq \twoheadrightarrow$ and $\twoheadleftarrow_n \cdot \Rightarrow \subseteq (\rightarrow^* \cdot \Rightarrow \cdot \twoheadleftarrow) \cup \twoheadleftarrow_{n-1}$.*

1. *Assume that $x \twoheadrightarrow y$ and $\mathsf{SN}(y, \Rightarrow)$. Then $\mathsf{SN}(x, \Rightarrow)$.*
2. *Assume that $\mathsf{WN}(x, \rightarrow)$, and for all y, if $y \in \mathsf{NF}(\rightarrow)$ then $\mathsf{SN}(y, \Rightarrow)$. Then $\mathsf{SN}(x, \Rightarrow)$.*

Proof. Part 1. Assume $\mathsf{SN}(y, \Rightarrow)$. Then $\Rightarrow^+$ is well-founded on successors of y. By induction on y_0 (for all y_0 with $y \Rightarrow^* y_0$) ordered by $\Rightarrow^+$, we will prove: $\forall n \forall x_0 : x_0 \twoheadrightarrow_n y_0 \Rightarrow \mathsf{SN}(x_0, \Rightarrow)$. The latter is proved by induction on n (inner induction). So assume $x_0 \twoheadrightarrow_n y_0$. It suffices to prove $\mathsf{SN}(z, \Rightarrow)$ for all z with $x_0 \Rightarrow z$. The main assumption gives two cases. In the first case, $y_0 \rightarrow^* \cdot \Rightarrow \cdot \twoheadleftarrow z$. In particular, as $\rightarrow \subseteq \Rightarrow$, we find y_1, such that $y_0 \Rightarrow^+ y_1$ and $z \twoheadrightarrow y_1$. By the outer induction hypothesis, $\mathsf{SN}(z, \Rightarrow)$. In the second case, $z \twoheadrightarrow_{n-1} y_0$. Then, by the inner induction hypothesis, $\mathsf{SN}(z, \Rightarrow)$.

Part 2. By $\mathsf{WN}(x, \rightarrow)$, we find a reduction $x \rightarrow^* y$, for some $y \in \mathsf{NF}(\rightarrow)$. Hence by using the assumptions, $x \twoheadrightarrow^* y$ and $\mathsf{SN}(y, \Rightarrow)$. By induction on the length of this reduction sequence and by Part 1, we obtain $\mathsf{SN}(x, \Rightarrow)$. $\square$

From now on, we assume a fixed TRS with a partial-order $>$ on rules, such that at least any two root-overlapping rules are comparable. Note in particular that for non-root-overlapping systems, taking the empty partial order is allowed. We now first define innermost rewriting with priority, which in case of overlaps gives priority to the smallest rule.

Definition 14. *A term t rewrites* innermost with priority *to a term u, notation $t \rightarrow_{i>} u$, if*

- *$t = C[\ell\sigma]$ and $u = C[r\sigma]$ for a rule $\ell \rightarrow r$, context C and substitution σ; and*
- *$\ell\sigma|_p \in \mathsf{NF}$ for all $p \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$; and*
- *there is no rule $(\ell' \rightarrow r')$ with $(\ell' \rightarrow r') < (\ell \rightarrow r)$ and substitution τ, such that $\ell'\tau = \ell\sigma$.*

Innermost reductions with priority in which $C[]$ is the empty context are called top innermost reductions*, notation $t \mapsto_{i>} u$.*

The first two conditions ensure innermost behavior and the third condition enforces the priority restrictions. Next, we define the generalized innermost rewriting with priority:

Definition 15. *A term t rewrites* generalized innermost with priority *to a term u, notation $t \rightarrow_{g>} u$, if*

- *$t = C[\ell\sigma]$ and $u = C[r\sigma]$ for a rule $\ell \rightarrow r$, context C and substitution σ; and*
- *$\ell\sigma|_p \in \mathsf{NF}$ for all $p \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$ such that $(\ell)_p \in \mathsf{Def}$; and*
- *there is no rule $(\ell' \rightarrow r')$ with $(\ell' \rightarrow r') < (\ell \rightarrow r)$ and substitution τ, such that $\ell'\tau = \ell\sigma$; and*
- *for all rules $(\ell' \rightarrow r')$ with $(\ell' \rightarrow r') < (\ell \rightarrow r)$ with the same top symbol f, and for all $1 \leq i \leq \mathsf{arity}(f)$ if $\ell\sigma|_i \notin \mathsf{NF}$ then $\ell'|_i \in \mathsf{LinVar}(\ell')$.*

Top generalized innermost reductions with priority in which $C[]$ is the empty context are denoted by $t \mapsto_{g>} u$.

The first two clauses ensure generalized innermost behavior, the third clause enforces priority restrictions, and the last technical clause ensures that doing some (innermost) steps doesn't influence which rule is chosen. (Adding a similar last clause to the definition of $\rightarrow_{i>}$ would not give a different relation.)

Note that $\rightarrow_{i>} \subseteq \rightarrow_{g>} \subseteq \rightarrow$. Moreover, if $>$ is well-founded, then $\mathsf{NF}(\rightarrow_{i>}) = \mathsf{NF}(\rightarrow_{g>}) = \mathsf{NF}(\rightarrow)$. Note that for non-root-overlapping systems, by choosing $> = \emptyset$, the last two conditions can be removed. So in this case, $\rightarrow_{g>} = \rightarrow_{g}$.

Top generalized innermost reduction with priority is deterministic in the following sense:

Lemma 16. *Assume that root-overlapping rules are comparable. If $t \mapsto_{g>} u$ and $t \mapsto_{g>} v$, then $u = v$.*

Proof. By definition of $\mapsto_{g>}$, we find rules $\ell_1 \rightarrow r_1$ and $\ell_2 \rightarrow r_2$ and substitutions σ and ρ, such that $t = \ell_1\sigma$ and $t = \ell_2\rho$. Then these rules have root-overlap, so they are comparable. By the third condition of Definition 15, $(\ell_1 \rightarrow r_1) \not> (\ell_2 \rightarrow r_2)$ and $(\ell_2 \rightarrow r_2) \not> (\ell_1 \rightarrow r_2)$, so $\ell_1 = \ell_2$ and $r_1 = r_2$. So σ and τ coincide on the variables in ℓ_1, so in particular on variables in r_1, so $u = r_1\sigma = r_2\tau = v$. □

Note that $\mapsto_{i>} \subseteq \mapsto_{g>}$, so the lemma also holds when one or both reduction steps are replaced by $\mapsto_{i>}$. One can prove that $\rightarrow_{i>}$ is confluent, because it satisfies the diamond property, and $\rightarrow_{g>}$ is weakly confluent, but not confluent, due to possible non-left-linearity.

We will also need parallel $\rightarrow_{i>}$-reduction:

Definition 17. *t rewrites to u with* n-step parallel innermost reduction with priority *(notation $t \twoheadrightarrow^{n}_{i>} u$) if $t = C[t_1, \ldots, t_n]$ and $u = C[t'_1, \ldots, t'_n]$ and for each j with $1 \leq j \leq n$, $t_j \mapsto_{i>} t'_j$. With $\twoheadrightarrow_{i>}$ we denote $\bigcup_{n\geq 0} \twoheadrightarrow^{n}_{i>}$.*

Note that $\twoheadrightarrow^{n}_{i>}$ doesn't coincide with the n-fold composition of $\twoheadrightarrow_{i>}$, as in $(\twoheadrightarrow_{i>})^n$.

Next we need an operation for simultaneous replacement of subterms.

Definition 18. *Let $t_1, \ldots, t_n, t'_1, \ldots, t'_n$ be given, such that $t_j \mapsto_{i>} t'_j$ (for all $1 \leq j \leq n$). We define the operation α on terms and extend it to substitutions as follows:*

- *for a term t the term $\alpha(t)$ is obtained from t by simultaneously replacing all occurrences of t_j by t'_j (for all j);*
- *$\alpha(\sigma)(x) = \alpha(x\sigma)$ for all variables x and all substitutions σ.*

Note that this is uniquely defined due to Lemma 16

We use the following facts on α and $\twoheadrightarrow_{i>}$:

Lemma 19. *Let $t_1, \ldots, t_n, t'_1, \ldots, t'_n$ be given, such that $t_j \mapsto_{i>} t'_j$ (for $1 \leq j \leq n$).*

1. *$t\sigma \twoheadrightarrow\!\!\!\!\!\!+_{i>} t^{\alpha(\sigma)}$; in particular, if $t\sigma \in \mathsf{NF}$ then $t^{\alpha(\sigma)} = t^{\sigma}$.*
2. *Assume that for all $p \in \mathsf{Pos}(\ell)$ with $(\ell)|_p \notin \mathcal{V}$, we have $\ell\sigma|_p \notin \{t_1, \ldots, t_n\}$. Then $\alpha(\ell^{\sigma}) = \ell^{\alpha(\sigma)}$.*

Proof. 1: All t_j are innermost redexes, so they occur at disjoint positions.
2: Induction on ℓ. □

We now prove the key lemma of this section:

Lemma 20. *Assume that any root-overlapping rules in the TRS are comparable in the partial order $>$. Then*

$$\twoheadleftarrow\!\!\!\!\!\!+^{n}_{i>} \cdot \rightarrow_{g>} \subseteq (\rightarrow^{*}_{i>} \cdot \rightarrow_{g>} \cdot \twoheadleftarrow\!\!\!\!\!\!+_{i>}) \cup \twoheadleftarrow\!\!\!\!\!\!+^{n-1}_{i>} \ .$$

Proof. Let $x \twoheadrightarrow\!\!\!\!\!\!+^{n}_{i>} y$ and $x \rightarrow_{g>} z$. Then $x|_p = \ell\sigma$ and $z = x[r\sigma]_p$ for certain $\ell \rightarrow r$, σ, and p; and $\ell^{\sigma} \mapsto_{g>} r^{\sigma}$, so, writing f for the top symbol of ℓ, we have the conditions of Definition 15:

C2 $\ell\sigma|_q \in \mathsf{NF}$ for all $q \in \mathsf{Pos}(\ell) \setminus \{\epsilon\}$ such that $(\ell)_q \in \mathsf{Def}$; and
C3 there is no rule $(\ell' \rightarrow r')$ with $(\ell' \rightarrow r') < (\ell \rightarrow r)$ and substitution τ, such that $\ell'\tau = \ell\sigma$; and
C4 for all rules $(\ell' \rightarrow r')$ with $(\ell' \rightarrow r') < (\ell \rightarrow r)$ with the same top symbol f, and for all $1 \leq i \leq \mathsf{arity}(f)$, either $\ell'|_i \in \mathsf{LinVar}(\ell')$, or $\ell\sigma|_i \in \mathsf{NF}$.

Moreover, we find pairwise disjoint positions $p_1, \ldots, p_n$, terms $t_1, \ldots, t_n, t'_1, \ldots, t'_n$, such that $x|_{p_j} = t_j$ and $t_j \mapsto_{i>} t'_j$, for all $1 \leq j \leq n$. Note that p cannot be strictly below any p_j, because t_j are innermost redexes.

We distinguish cases. Case 1: $\ell^{\sigma} = t_j$ for some $1 \leq j \leq n$. By Lemma 16, $r^{\sigma} = t'_j$. Now if $p = p_j$, for some $1 \leq j \leq n$, we can write x as $C[t_1, \ldots, t_n]$, and prove the lemma as follows:

$$z = x[r\sigma]_p = C[t_1, \ldots, t'_j, \ldots, t_n] \twoheadrightarrow\!\!\!\!\!\!+^{n-1}_{i>} C[t'_1, \ldots, t'_n] = y$$

Otherwise, p is disjoint from all p_j, so we can write x as $C[l^{\sigma}, t_1, \ldots, t_n]$, and prove the lemma as follows:

$$y = C[l^{\sigma}, t'_1, \ldots, t'_n] \rightarrow_{g>} C[r^{\sigma}, t'_1, \ldots, t'_n] \twoheadleftarrow\!\!\!\!\!\!+^{n}_{i>} C'[r^{\sigma}, t_1, \ldots, t_n] = z$$

Case 2: for all j, $\ell^{\sigma} \neq t_j$. Then all p_j are either disjoint from p, or strictly below p. Assume (w.l.o.g.) that positions $p_1, \ldots, p_k$ are strictly below p, and $p_{k+1}, \ldots, p_n$ are disjoint from p (for some $0 \leq k \leq n+1$). So we find context $C[]$ and $D[]$ such that $x = C[l\sigma, t_{k+1}, \ldots, t_n]$ and $\ell\sigma = D[t_1, \ldots, t_k]$. We can write $y = C[D[t'_1, \ldots, t'_k], t'_{k+1}, \ldots, t'_n]$ and $z = C[r^{\sigma}, t_{k+1}, \ldots, t_n]$. In order to apply $\ell \rightarrow r$ to y, we first have to reduce any remaining copies of t_j in $D[t'_1, \ldots, t'_k]$ (in case ℓ is non-left-linear). This is done by the simultaneous replacement α, defined in Definition 18 for terms $t_1, \ldots, t_k$.

Define $v := C[\alpha(\ell^{\sigma}), t'_{k+1}, \ldots, t'_n]$. We next show that $\alpha(\ell^{\sigma})$ is an instance of ℓ, using Lemma 19.2. So let $q \in \mathsf{Pos}(\ell)$, $(\ell)|_q \notin \mathcal{V}$, and assume $\ell\sigma|_q = t_j$ (for some $0 \leq j \leq k$). Then $(\ell\sigma)_q \in \mathsf{Def}$, so $(\ell)_q \in \mathsf{Def}$ (it is not in $\mathcal{V}$ by assumption). Note that $q \neq \varepsilon$ (by assumption of Case 2). Hence, by C2, $\ell\sigma|_q \in \mathsf{NF}$, in contradiction with t_j being an innermost redex. So by Lemma 19.2, we indeed get that $\alpha(\ell^\sigma) = \ell^{\alpha(\sigma)}$.

So we proved $v = C[l^{\alpha(\sigma)}, t'_{k+1}, \dots, t'_n]$. Define $w := C[r^{\alpha(\sigma)}, t'_{k+1}, \dots, t'_n]$. Then by Lemma 19.1, $\ell^\sigma \twoheadrightarrow\!\!\!\!\!\shortmid_{i>} \ell^{\alpha(\sigma)} = \alpha(\ell^\sigma)$. Then also $D[t'_1, \dots, t'_k] \twoheadrightarrow\!\!\!\!\!\shortmid_{i>} \alpha(\ell^\sigma)$ (this contracts a subset of the redexes from ℓ^σ). Then also $y \twoheadrightarrow\!\!\!\!\!\shortmid_{i>} v$, as $\twoheadrightarrow\!\!\!\!\!\shortmid_{i>}$ is closed under context. Similarly, $r^\sigma \twoheadrightarrow\!\!\!\!\!\shortmid_{i} r^{\alpha(\sigma)}$ by Lemma 19.1, so $z \twoheadrightarrow\!\!\!\!\!\shortmid_{i} w$, because $p_{k+1}, \dots, p_n$ are all disjoint from p.

Finally, we must check that $\ell^{\alpha(\sigma)} \mapsto_{g>} r^{\alpha(\sigma)}$, in order to conclude that $v \to_{g>} w$. This boils down to checking conditions 2–4 of Definition 15, assuming that these conditions (called C2–C4) hold for $\ell^\sigma \mapsto_{g>} r^\sigma$.

Cond 2) Let $q \in \mathsf{Pos}(\ell) \setminus \{\varepsilon\}$ with $(l)_q \in \mathsf{Def}$. Then (using equations of Lemma 19.1, 19.2 and usual commutation of substitutions and positions), we have $\ell^{\alpha(\sigma)}|_q = \alpha(\ell^\sigma|_q) = \ell^\sigma|_q \in \mathsf{NF}$ by C2.

Cond 3) Let $(\ell' \to r') < (\ell \to r)$ and τ be given, such that $\ell'\tau = l^{\alpha(\sigma)}$. We will construct τ', such that $\ell'\tau' = l^\sigma$, contradicting C3, as follows: $\tau'(x) :=$ **if** $x = \ell'|_i \in \mathsf{LinVar}(\ell')$ for some $0 \leq i \leq \mathsf{arity}(f)$, **then** $\ell|_i\sigma$; **else** $\tau(x)$. Then for each $0 \leq i \leq \mathsf{arity}(f)$, by C4, either $\ell'|_i \in \mathsf{LinVar}(\ell')$, so $\ell'|_i\tau' = \ell|_i\sigma$ by definition of τ', or $\ell\sigma|_i \in \mathsf{NF}$, so $\ell|_i\sigma = \ell|_i^{\alpha(\sigma)} = \ell'|_i\tau = \ell'|_i\tau'$. This holds for all arguments i, hence $\ell'\tau' = l^\sigma$, contradicting C3.

Cond 4) Let $(\ell' \to r') < (\ell \to r)$. By C4, either $\ell\sigma|_i \in \mathsf{NF}$, hence also $\ell^{\alpha(\sigma)}|_i = \ell\sigma|_i \in \mathsf{NF}$, or $\ell'|_i \in \mathsf{LinVar}(\ell')$.

Summarizing, we obtain:

$$\begin{aligned} y &= C[D[t'_1, \dots, t'_k], t'_{k+1}, \dots, t'_n] \\ \twoheadrightarrow\!\!\!\!\!\shortmid_{i>}\ v &= C[\alpha(l^\sigma), t'_{k+1}, \dots, t'_n] \\ &= C[l^{\alpha(\sigma)}, t'_{k+1}, \dots, t'_n] \\ \to_{g>}\ w &= C[r^{\alpha(\sigma)}, t'_{k+1}, \dots, t'_n] \\ \twoheadleftarrow\!\!\!\!\!\shortmid_{i>}\ z &= C[r^\sigma, t_{k+1}, \dots, t_n] \end{aligned}$$

which proves the lemma, by observing that $\twoheadrightarrow\!\!\!\!\!\shortmid_{i>} \subseteq \to^*_{i>}$. □

Theorem 21. *Assume that any root-overlapping rules in the TRS are comparable by the partial order $>$. Then for any term t, if $\mathsf{WN}(t, \to_{i>})$, then $\mathsf{SN}(t, \to_{g>})$.*

Proof. Assume $\mathsf{WN}(t, \to_{i>})$. We check the conditions of Lemma 13.2, with $\Rightarrow\ = \ \to_{g>}$, $\to\ =\ \to_{i>}$ and $\twoheadrightarrow\!\!\!\!\!\shortmid_n\ =\ \twoheadrightarrow\!\!\!\!\!\shortmid^n_{i>}$. Clearly, $\to_{i>} \subseteq \to_{g>}$ and $\to_{i>}\ =\ \twoheadrightarrow\!\!\!\!\!\shortmid^1_{i>} \subseteq \twoheadrightarrow\!\!\!\!\!\shortmid_{i>}$. The next condition is obtained from Lemma 20. Finally, $\to_{i>}$-normal forms are $\to_{g>}$-normal forms, so in particular $\to_{g>}$-terminating. Hence we can apply Lemma 13.2, and obtain $\mathsf{SN}(t, \to_{g>})$ □

Note that if we drop the third condition of Definition 15, the previous theorem would not hold. This is witnessed by Example 4. Also the fourth condition is essential, as witnessed by the following example. Consider the TRS consisting of the rules

$$e \to a, \quad d \to d, \quad \alpha : f(x, x) \to c, \quad \beta : f(a, y) \to d$$

with $\alpha < \beta$. Then $f(a,e) \rightarrow_{i>} f(a,a) \rightarrow_{i>} c$ is the only $\rightarrow_{i>}$-reduction from $f(a,e)$, so $\mathsf{SN}(f(a,e), \rightarrow_{i>})$. However, without the fourth condition of Definition 15 we would have $f(a,e) \rightarrow_{g>} d \rightarrow_{g>} d$, leading to an infinite reduction. A similar left-linear example exists:

$$e \rightarrow b, \quad d \rightarrow d, \quad \alpha : f(x,b) \rightarrow c, \quad \beta : f(a,y) \rightarrow d$$

with again $\alpha < \beta$. Now $\mathsf{SN}(f(a,e), \rightarrow_{i>})$, but without the fourth condition of Definition 15, we would obtain $f(a,e) \rightarrow_{g>} d \rightarrow_{g>} d$, leading to an infinite reduction. The following corollary shows that generalized innermost rewriting with priority is not worse than usual innermost rewriting.

Corollary 22. *Assume that any root-overlapping rules in the TRS are comparable by the partial order $>$. Let t be a term.*

- $\mathsf{SN}(t, \rightarrow_{i>})$ *if and only if* $\mathsf{SN}(t, \rightarrow_{g>})$.
- *If* $\mathsf{SN}(t, \rightarrow_{i})$*, then* $\mathsf{SN}(t, \rightarrow_{g>})$.

Proof. This follows from Theorem 21 using $\rightarrow_{i>} \subseteq \rightarrow_{i}$ and $\rightarrow_{i>} \subseteq \rightarrow_{g>}$. □

The reverse of the second doesn't hold, as witnessed by the two rules $(a \rightarrow b) < (a \rightarrow a)$. The infinite reduction $a \rightarrow_{i} a$ is disabled in $\rightarrow_{g>}$ by the terminating smaller rule.

Corollary 23. *Let the TRS be non-root-overlapping and let t be a term. Then* $\mathsf{WN}(t, \rightarrow_{i}) \Leftrightarrow \mathsf{SN}(t, \rightarrow_{g}) \Leftrightarrow \mathsf{SN}(t, \rightarrow_{i})$.

Proof. Assume $\mathsf{WN}(t, \rightarrow_{i})$. The TRS is non-root overlapping, so we can take $> = \emptyset$, then $\rightarrow_{i} = \rightarrow_{i>}$ and $\rightarrow_{g} = \rightarrow_{g>}$. By Theorem 21, we obtain $\mathsf{SN}(t, \rightarrow_{g})$. The implication $\mathsf{SN}(t, \rightarrow_{g}) \Rightarrow \mathsf{SN}(t, \rightarrow_{i})$ follows from $\rightarrow_{i} \subseteq \rightarrow_{g}$; the implication $\mathsf{SN}(t, \rightarrow_{i}) \Rightarrow \mathsf{WN}(t, \rightarrow_{i})$ is universal. □

In [10] the conjecture was stated that if a TRS is innermost terminating, then any in-time JITty annotation induces a terminating strategy. A JITty annotation for a function symbol is a list consisting of argument positions and rules for that symbol, which deterministically describes in which order to evaluate the arguments or apply the rules. An annotation induces a rewrite relation $\rightarrow_{strat}$. E.g., given rules $\alpha : \mathsf{if}(\mathsf{true}, x, y) \rightarrow x$ and $\beta : \mathsf{if}(\mathsf{false}, x, y) \rightarrow y$, the annotation $\mathsf{if} : [1, \alpha, \beta, 2, 3]$ denotes that $\mathsf{if}(s, t, u)$ is evaluated by first evaluating s, then trying rule α, then β, and if this failed, normalize t and u, respectively. A strategy annotation is *in-time* if for every rule $\ell \rightarrow r$ in it, all argument positions in ℓ distinct from $\mathsf{LinVar}(\ell)$ occur before it. We can now solve this conjecture.

Corollary 24. *Let the TRS be finite, and let strat be an in-time strategy annotation in the sense of [10]. For all terms t, if* $\mathsf{SN}(t, \rightarrow_{i})$ *then* $\mathsf{SN}(t, \rightarrow_{strat})$.

Proof. Define $(\ell \rightarrow r) < (\ell' \rightarrow r')$ if and only if ℓ and ℓ' have the same top symbol, and $\ell \rightarrow r$ occurs before $(\ell' \rightarrow r')$ in the strategy annotation. Then $\rightarrow_{strat} \subseteq \rightarrow_{g>}$; conditions 2 and 4 of Definition 15 are enforced by the in-time requirement, and condition 3 is enforced because the order coincides with the order in the annotation. Assume $\mathsf{SN}(t, \to_i)$. Note that $\to_{i>} \subseteq \to_i$ and they have the same normal forms, so $\mathsf{WN}(t, \to_{i>})$. By Theorem 21, $\mathsf{SN}(t, \to_{g>})$, hence $\mathsf{SN}(t, \to_{strat})$. □

The last result doesn't hold for all (eager) OBJ annotations, in which 0 is used to denote application of any rule. Consider again the example TRS consisting of the following three rules $\alpha : f(a, b, x) \to f(x, x, x)$, $\beta : c \to a$, $\gamma : c \to b$. The system is innermost terminating, so any JITty annotation gives a terminating strategy, including $f : [1, 2, \alpha, 3]$, and either $c : [\beta, \gamma]$ or $c : [\gamma, \beta]$. However, the OBJ-annotation $f : [1, 2, 0, 3]$ and $c : [0]$ (where 0 stands for the application of any rule) admits an infinite sequence.

6 Conclusions

We introduced two generalizations $\to_{ndg}$ and $\to_{g>}$ of innermost rewriting $\to_i$ and $\to_{i>}$, respectively, for which we proved that for every term t the properties $\mathsf{SN}(t, \to_i)$ and $\mathsf{SN}(t, \to_{ndg})$ are equivalent, and the properties $\mathsf{SN}(t, \to_{i>})$ and $\mathsf{SN}(t, \to_{g>})$ are equivalent. As a main application of these results we see that for particular strategies as they are applied in implementations ([4, 11]) we may conclude that they are not worse than innermost rewriting as long as they are contained in $\to_{ndg}$ or $\to_{g>}$. This comparison describes worst case behavior; in typical applications we observe that the particular strategies terminate where innermost rewriting does not. Roughly speaking we can say that these strategies allow a kind of lazy rewriting without loss of efficiency or termination behavior.

We want to emphasize that these strategies apply for all TRSs without any restriction, and have the same set of normal forms as the full general rewrite relation, in contrast to other approaches like context-sensitive rewriting ([6]). Moreover, our strategies do not depend on user-defined options, except for the order of root-overlapping rules in Section 5.

In implementations typically strategies are deterministic. For a proper order $>$ on the rules and $\to \in \{\to_i, \to_{i>}, \to_{g>}, \to_{ndg}\}$ let $\stackrel{d}{\to}$ be a deterministic instance of $\to$, i.e., $\stackrel{d}{\to} \subseteq \to$ and for every term t not being a normal form there is exactly one u satisfying $t \stackrel{d}{\to} u$. Then for every term t we have the following properties:

$$
\begin{array}{ccccccc}
\mathsf{SN}(t, \to_{ndg}) & \Leftrightarrow & \mathsf{SN}(t, \to_i) & \Rightarrow & \mathsf{SN}(t, \to_{i>}) & \Leftrightarrow & \mathsf{SN}(t, \to_{g>}) \\
\Downarrow & & \Downarrow & & \Updownarrow & & \Downarrow \\
\mathsf{SN}(t, \stackrel{d}{\to}_{ndg}) & & \mathsf{SN}(t, \stackrel{d}{\to}_i) & & \mathsf{SN}(t, \stackrel{d}{\to}_{i>}) & & \mathsf{SN}(t, \stackrel{d}{\to}_{g>}) \\
\Updownarrow & & \Updownarrow & & \Updownarrow & & \Updownarrow \\
\mathsf{WN}(t, \stackrel{d}{\to}_{ndg}) & & \mathsf{WN}(t, \stackrel{d}{\to}_i) & & \mathsf{WN}(t, \stackrel{d}{\to}_{i>}) & & \mathsf{WN}(t, \stackrel{d}{\to}_{g>}) \\
\Downarrow & & \Downarrow & & \Updownarrow & & \Downarrow \\
\mathsf{WN}(t, \to_{ndg}) & \Leftarrow & \mathsf{WN}(t, \to_i) & \Leftarrow & \mathsf{WN}(t, \to_{i>}) & \Rightarrow & \mathsf{WN}(t, \to_{g>})
\end{array}
$$

In this diagram the equivalences in the first line are Theorem 9 and Corollary 22; the vertical equivalences involving $\to_{i>}$ follow from Theorem 21. All other implications and equivalences are immediate from the definitions. For none of the implications the converse holds as is easily checked by considering the term a w.r.t. the two rules $a \to a, a \to b$ and the term $f(a)$ w.r.t. the two rules $a \to a, f(x) \to b$, for various orders of the rules and deterministic instances. A remaining question is whether $\mathsf{WN}(t, \to_{ndg})$ and $\mathsf{WN}(t, \to_{g>})$ are comparable. They are not, as follows from the following examples. Let the TRS consist of the rules $a \to a$, $f(x) \to g(x,x)$ and $g(x,y) \to b$ and let $>$ be empty. Then we have $\mathsf{WN}(f(a), \to_{g>})$ but not $\mathsf{WN}(f(a), \to_{ndg})$. Conversely, let the TRS consist of the rules $f(x) \to g(x)$, $f(x) \to h(x)$, $g(g(x)) \to g(g(x))$ and $h(h(x)) \to h(h(x))$. Then $f(f(x)) \to_i^+ g(h(x))$ and $f(f(x)) \not\to_{g>}^+ g(h(x))$ for any order $>$. Hence both $\mathsf{WN}(f(f(x)), \to_i)$ and $\mathsf{WN}(f(f(x)), \to_{ndg})$ hold, while for no order $>$ on the rules $\mathsf{WN}(f(f(x)), \to_{g>})$ holds.

For non-root-overlapping TRSs we proved that termination of $\to_g$ and $\to_i$ are equivalent. As an open question we leave whether in this claim the condition of being non-root-overlapping can be weakened to confluence.

Acknowledgment. We like to thank Vincent van Oostrom for fruitful discussions and suggestions for improvement.

References

1. F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambr. Univ. Pr., 1998.
2. J.C.M. Baeten, J.A. Bergstra, J.W. Klop, and W.P. Weijland. Term rewriting systems with rule priorities. *Theoretical Computer Science*, 67:283–301, 1989.
3. W.J. Fokkink, J.F.Th. Kamperman, and H.R. Walters. Lazy rewriting and eager machinery. *ACM Transactions on Programming Languages and Systems*, 22(1):45–86, 2000.
4. J. Goguen, T. Winkler, J. Meseguer, K. Futatsugi, and J.-P. Jouannaud. Introducing OBJ. In J. Goguen and G. Malcolm, editors, *Software Engineering with OBJ: algebraic specification in action*. Kluwer, 2000.
5. B. Gramlich. Abstract relations between restricted termination and confluence properties of rewrite systems. *Fundamenta Informaticae*, 24:3–23, 1995.
6. S. Lucas. Context-sensitive rewriting strategies. *Information and Computation*, 178:294–343, 2002.
7. M. Nakamura and K. Ogata. The evaluation strategy for head normal form with and without on-demand flags. In K. Futatsugi, editor, *The 3rd Int. W. on Rewriting / Logic and its Applications (WRLA2000)*, volume 36 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2001.
8. M. J. O'Donnell. *Computing in systems described by equations*, volume 58 of *Lecture Notes in Computer Science*. Springer, 1977.
9. Y. Toyama. Counterexamples to the termination for the direct sum of term rewriting systems. *Information Processing Letters*, 25:141–143, 1987.
10. J. C. van de Pol. Just-in-time: on strategy annotations. In *Int. Workshop on Reduction Strategies in Rewriting and Programming (WRS 2001)*, volume 57 of *Electronic Notes in Theoretical Computer Science*, 2001.
11. J. C. van de Pol. JITty: a rewriter with strategy annotations. In *Proc. 13th RTA*, volume 2378 of *Lecture Notes in Computer Science*, pages 367–370, 2002.