Call by Need Computations to Root-Stable Form

Aart Middeldorp

Institute of Information Sciences and Electronics
University of Tsukuba
Tsukuba 305, Japan
ami@score.is.tsukuba.ac.jp

Abstract

The following theorem of Huet and Lévy forms the basis of all results on optimal reduction strategies for orthogonal term rewriting systems: every term not in normal form contains a needed redex, and repeated contraction of needed redexes results in the normal form, if the term under consideration has one. We generalize this theorem to computations to root-stable form and we argue that the resulting notion of root-neededness is more fundamental than (other variants of) neededness when it comes to infinitary normalization.

1 Introduction

In this paper we are concerned with reduction strategies for term rewriting systems. A reduction strategy is called normalizing if repeated contraction of the redexes selected by the strategy leads to normal form. O'Donnell [13] showed that the parallel-outermost strategy, which contracts all outermost redexes in parallel, is normalizing for orthogonal term rewriting systems. Parallel-outermost is not an optimal reduction strategy since many of the redex contractions it performs are useless for obtaining normal forms. Optimal reduction strategies for orthogonal term rewriting systems are invariably based on the following landmark theorem of Huet and Lévy [6]: (1) every term not in normal form contains a *needed* redex and (2) repeated contraction of needed redexes results in the normal form, if the term under consideration has one. A needed redex is a redex (a descendant of) which is contracted in every rewrite sequence to normal form. So needed redexes have to be evaluated in order to obtain normal forms.

This paper appeared in the Proceedings of the 24th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, Paris, pp. 94–105, January 1997.

© 1997 ACM 0-89791-853-3/96/01

By definition every redex in a term that has no normal form is needed. Hence the theory of needed reduction is not useful for terms that have an infinite normal form, a common situation in functional programming languages with lazy semantics. Consider for instance the following rewrite rules implementing the Sieve of Eratosthenes for generating the infinite list of all prime numbers:

```
primes          →  sieve(from(2))
from(x)         →  x:from(x+1)
sieve(x:y)      →  x:sieve(filter(x,y))
filter(x,y:z)   →  if(x|y,filter(x,z),
                          y:filter(x,z))
if(true,x,y)    →  x
if(false,x,y)   →  y
```

Assuming that addition (`+`) is an invisible built-in operation, the evaluation of `primes` starts as follows:

```
primes  →  sieve(from(2))
        →  sieve(2:from(3))
```

At this point there are two ways to proceed. Either we contract `sieve(2:from(3))` using the third rewrite rule, resulting in `2:sieve(filter(2,from(3)))`, or we contract the redex `from(3)` using the second rewrite rule, resulting in `sieve(2:(3:from(4)))`. Because the term `sieve(2:from(3))` doesn't have a (finite) normal form, both redexes are trivially needed. Consequently, the theory of needed reduction cannot distinguish the meaningless infinite computation

```
primes  →*  sieve(2:from(3))
        →   sieve(2:(3:from(4)))
        →   sieve(2:(3:(4:from(5))))
        →   ···
```

from the infinite computation

```
primes  →*  2:sieve(2:from(3))
        →   2:sieve(filter(2,from(3)))
        →*  2:(3:sieve( ... ))
        →   ···
```

whose limit is the infinite list

```
2:(3:(5:(7: ... )))
```

of prime numbers.

In this paper we develop the theory of *root-needed* reduction. A root-needed redex is a redex (a descendant of) which is contracted in every rewrite sequence to root-stable form. For instance, redex `from(3)` in `sieve(2:from(3))` is not root-needed because by contracting the other redex `sieve(2:from(3))` we obtain the term `2:sieve(filter(2,from(3)))` which is root-stable as its root symbol `:` is a constructor.

It is well-known that in the λ-calculus the leftmost-outermost redex in a term that is not yet root-stable is root-needed, but it is undecidable whether an arbitrary redex is root-needed (cf. Barendregt *et al.* [3] where these statements are proved for the related notion of neededness with respect to head-normal forms). For orthogonal term rewriting things are more complicated because the leftmost-outermost redex is in general not root-needed. Nevertheless, we will prove that every non-root-stable term has a root-needed redex. The first result of Huet and Lévy cited above is an easy consequence of this result. Our proof is much simpler than the one in [6]. Our second result is the root-normalization of root-needed reduction, which we obtain by a modification of the proof of Sekar and Ramakrishnan [15] of an extension of the second result of Huet and Lévy cited above.

After these two fundamental results we investigate the relationship between root-normalization and normalization of reduction strategies. These two concepts are not equivalent, but we identify a natural property, *context-freeness*, under which root-normalization implies normalization. Since root-normalization of a reduction strategy is a much weaker proof obligation than normalization, this may lead to a simplification of the existing theory on normalizing reduction strategies. More importantly, we will show that under a natural fairness requirement context-free root-normalizing reduction strategies are *infinitary normalizing*. Since a similar statement does not hold for normalizing reduction strategies, this clearly shows that root-normalization rather than normalization is the key property of reduction strategies.

The remainder of this paper is organized as follows. In the next section we recall well-known results for orthogonal and almost orthogonal term rewriting. Root-stability is introduced in Section 3. In Section 4 we prove that every non-root-stable term in an orthogonal term rewriting system has a root-needed redex. In Section 5 we present our root-normalization results. Root-normalization and normalization of reduction strategies are compared in Section 6. In Section 7 we show that under natural assumptions root-normalization implies infinitary normalization. We compare our results with the infinitary normalization results in Kennaway *et al.* [9]. We conclude in Section 8 with a few remarks on (un)decidability.

2 Preliminaries

In this preliminary section we recall the basic notions of term rewriting. Further details can be found in [5, 11].

A signature is a set $\mathcal{F}$ of function symbols. Associated with every $f \in \mathcal{F}$ is a natural number denoting its arity. Function symbols of arity 0 are called constants. The set $\mathcal{T}(\mathcal{F},\mathcal{V})$ of terms built from a signature $\mathcal{F}$ and a countably infinite set of variables $\mathcal{V}$ disjoint from $\mathcal{F}$ is the smallest set containing $\mathcal{V}$ such that $f(t_1,\dots,t_n) \in \mathcal{T}(\mathcal{F},\mathcal{V})$ whenever $f \in \mathcal{F}$ has arity n and $t_1,\dots,t_n \in \mathcal{T}(\mathcal{F},\mathcal{V})$. The set of variables occurring in a term t is denoted by $\mathcal{V}\mathrm{ar}(t)$. A linear term doesn't contain multiple occurrences of the same variable. The root symbol of a term t is defined as follows: $\mathrm{root}(t) = t$ if t is a variable and $\mathrm{root}(t) = f$ if $t = f(t_1,\dots,t_n)$.

A position is a finite sequence of natural numbers identifying a subterm occurrence in a term. The empty or root position is denoted by ε and the concatenation of positions p and q is denoted by $p \cdot q$. The set $\mathcal{P}\mathrm{os}(t)$ of positions in a term t is inductively defined as follows: $\mathcal{P}\mathrm{os}(t) = \{\varepsilon\}$ if t is a variable and $\mathcal{P}\mathrm{os}(t) = \{\varepsilon\} \cup \{i \cdot p \mid 1 \leqslant i \leqslant n \text{ and } p \in \mathcal{P}\mathrm{os}(t_i)\}$ if $t = f(t_1,\dots,t_n)$. We say that a position p is above a position q if there exists a (necessarily unique) position r such that $p \cdot r = q$, in which case we define $q \backslash p$ as the position r. If p is above q we also say that q is below p and we write $p \leqslant q$. We write $p < q$ if $p \leqslant q$ and $p \neq q$. Positions p, q are disjoint, denoted by $p \perp q$, if neither $p \leqslant q$ nor $q \leqslant p$. If $p \in \mathcal{P}\mathrm{os}(t)$ then $t_{|p}$ denotes the subterm of t at position p and $t[s]_p$ denotes the term that is obtained from t by replacing the subterm at position p by the term s. The set $\mathcal{P}\mathrm{os}(t)$ is partitioned into $\mathcal{P}\mathrm{os}_{\mathcal{V}}(t) = \{p \in \mathcal{P}\mathrm{os}(t) \mid t_{|p} \in \mathcal{V}\}$ and $\mathcal{P}\mathrm{os}_{\mathcal{F}}(t) = \mathcal{P}\mathrm{os}(t) \setminus \mathcal{P}\mathrm{os}_{\mathcal{V}}(t)$.

A substitution is a map σ from $\mathcal{V}$ to $\mathcal{T}(\mathcal{F},\mathcal{V})$ with the property that its domain $\{x \in \mathcal{V} \mid \sigma(x) \neq x\}$ is finite. The result of applying the substitution σ to the term t is denoted by $t\sigma$. We say that $t\sigma$ is an instance of t.

A rewrite rule $l \to r$ is a pair of terms such that the left-hand side l is not a variable and variables which occur in the right-hand side r occur also in l. A term rewriting system (TRS for short) over a signature $\mathcal{F}$ is a set $\mathcal{R}$ of rewrite rules between terms in $\mathcal{T}(\mathcal{F},\mathcal{V})$. A TRS $\mathcal{R}$ defines a rewrite relation $\to_{\mathcal{R}}$ on terms as follows: $s \to_{\mathcal{R}} t$ if there exists a rewrite rule $l \to r$ in $\mathcal{R}$, a substitution σ, and a position $p \in \mathcal{P}\mathrm{os}(s)$ such that

$s_{|p} = l\sigma$ and $t = s[r\sigma]_p$. The subterm $l\sigma$ of s is called a redex and we say that s rewrites to t by contracting the redex $l\sigma$ at position p. The left-hand side l is called the pattern of the redex $l\sigma$. In general redexes may have more than one pattern. The subterm $r\sigma$ of t is called a contractum of the redex $l\sigma$ and we call $s \to_{\mathcal{R}} t$ a rewrite step. We often omit the subscript $\mathcal{R}$.

A term without redexes is called a normal form. The transitive reflexive closure $\to^*$ of the rewrite relation is called reduction or rewriting and its inverse ${}^*\!\leftarrow$ expansion. If $s \to^* t$ we say that t is a reduct of s. We say that s has a normal form (t) if it has a reduct (t) that is a normal form. A TRS is confluent if the inclusion ${}^*\!\leftarrow \cdot \to^* \subseteq \to^* \cdot {}^*\!\leftarrow$ holds. We also say that the relation $\to$ is confluent. In a confluent TRS every term has at most one normal form. We write $s \xrightarrow{\varepsilon} t$ if $s \to t$ by contracting the redex at the root position, so s is a redex and t a contractum of s. We write $s \xrightarrow{>\varepsilon} t$ if $s \to t$ by contracting a redex at a non-root position.

Lemma 2.1 *Let $\mathcal{R}$ be a confluent TRS. The relation $\xrightarrow{>\varepsilon}_{\mathcal{R}}$ is confluent.*

Proof. Suppose $s \xrightarrow{>\varepsilon}{}^* t_1$ and $s \xrightarrow{>\varepsilon}{}^* t_2$. We may write $s = f(s_1, \ldots, s_n)$, $t_1 = f(s_1^1, \ldots, s_n^1)$, and $t_2 = f(s_1^2, \ldots, s_n^2)$ with $s_i \to^* s_i^1$ and $s_i \to^* s_i^2$ for all $1 \leqslant i \leqslant n$. Confluence yields terms $s_1^3, \ldots, s_n^3$ such that $s_i^1 \to^* s_i^3$ and $s_i^2 \to^* s_i^3$ for every $1 \leqslant i \leqslant n$. Let $t_3 = f(s_1^3, \ldots, s_n^3)$. We clearly have $t_1 \xrightarrow{>\varepsilon}{}^* t_3$ and $t_2 \xrightarrow{>\varepsilon}{}^* t_3$. □

A TRS is left-linear if the left-hand sides of its rewrite rules are linear terms. A TRS is non-erasing if $\mathcal{V}\text{ar}(l) = \mathcal{V}\text{ar}(r)$ for every rewrite rule $l \to r$. Let $l_1 \to r_1$ and $l_2 \to r_2$ be renamed versions of rewrite rules of a TRS $\mathcal{R}$ such that they have no variables in common. Suppose $l_{1|p}$ for some $p \in \mathcal{P}\text{os}_{\mathcal{F}}(l_1)$ and l_2 are unifiable with most general unifier σ. The pair of terms $\langle l_1[r_2]_p\sigma, r_1\sigma\rangle$ is called a critical pair of $\mathcal{R}$. If $l_1 \to r_1$ and $l_2 \to r_2$ are renamed versions of the same rewrite rule, we do not consider the case $p = \varepsilon$. A critical pair $\langle l_1[r_2]_p\sigma, r_1\sigma\rangle$ with $p = \varepsilon$ is an overlay. A critical pair $\langle s, t\rangle$ is trivial if $s = t$. We are mostly concerned with *orthogonal* and *almost* orthogonal TRSs in this paper. An orthogonal TRS is a left-linear TRS without critical pairs. Every redex in an orthogonal TRS has exactly one pattern. A left-linear TRS with the property that all its critical pairs are trivial overlays is called almost orthogonal. The standard example of an almost orthogonal TRS that is not orthogonal is parallel-or:

$$\begin{aligned} \top \vee x &\to \top \\ x \vee \top &\to \top \end{aligned}$$

Almost orthogonal TRSs are confluent. Most results obtained for orthogonal TRSs (Klop [11] contains an overview) carry over to almost orthogonal TRSs with verbatim the same proofs. The notable exception is the theory of Huet and Lévy [6] on needed reduction which is explained below.

Let $A\colon s \to t$ by contracting the redex at position $p \in \mathcal{P}\text{os}(s)$ using the rewrite rule $l \to r$. Let $q \in \mathcal{P}\text{os}(s)$. The set $q\backslash A$ of *descendants* of q in t is defined as follows:

$$q\backslash A = \begin{cases} \{q\} & \text{if } q < p \text{ or } q \perp p, \\ \{p \cdot p_3 \cdot p_2 \mid r_{|p_3} = l_{|p_1}\} & \\ & \text{if } q = p \cdot p_1 \cdot p_2 \text{ with } p_1 \in \mathcal{P}\text{os}_{\mathcal{V}}(l), \\ \varnothing & \text{otherwise.} \end{cases}$$

If $Q \subseteq \mathcal{P}\text{os}(s)$ then $Q\backslash A$ denotes the set $\bigcup_{q \in Q} q\backslash A$. The notion of descendant extends to rewrite sequences in the obvious way. If Q is a set of pairwise disjoint positions in s and $A\colon s \to^* t$ then the positions in $Q\backslash A$ are pairwise disjoint. In the following we are only interested in descendants of redex positions. (These are often called *residuals* in the literature.) For convenience we identify redex positions with redexes. Almost orthogonal TRSs have the nice property that descendants of a redex are redexes (with the same patterns).

Definition 2.2 Let $\mathcal{R}$ be a TRS. A redex Δ in a term t is called *needed* if in every rewrite sequence from t to normal form a descendant of Δ is contracted.

Let $A\colon s \to^* t$ be a rewrite sequence and Δ a redex in s. We write $\Delta \perp A$ if no descendant of Δ is contracted in A. So a redex Δ in a term s is needed if and only if t is not a normal form whenever $A\colon s \to^* t$ with $\Delta \perp A$.

Theorem 2.3 (Huet and Lévy [6]) *Let $\mathcal{R}$ be an orthogonal TRS.*

(1) *Every term that is not a normal form has a needed redex.*
(2) *Repeated contraction of needed redexes results in a normal form, whenever the term under consideration has a normal form.*

□

In Section 4 we prove a proper extension of the first statement by a simple induction argument. The second statement is known as the *normalization* of needed reduction. Actually, in [6] the *hyper* normalization of needed reduction is proved: if a term t has a normal form then there are no rewrite sequences starting from t that contain infinitely many needed steps.

Sekar and Ramakrishnan [15] observed that Theorem 2.3(1) doesn't extend to almost orthogonal TRSs: neither of the two redexes in the term $(\top \vee \top) \vee (\top \vee \top)$ is needed with respect to the parallel-or TRS. They introduced the following concept in order to minimize rather than eliminate useless redex contractions.

Definition 2.4 Let $\mathcal{R}$ be a TRS. A non-empty set of redexes in a term t is called *necessary* if in every rewrite

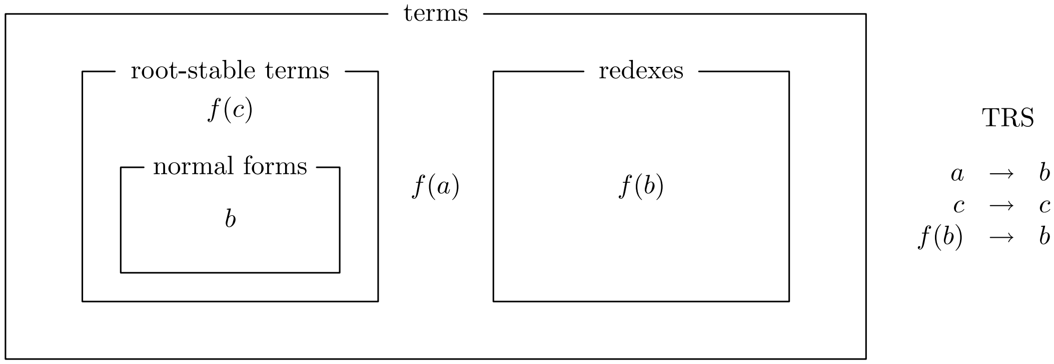


Figure 1: Root-stable terms, normal forms, and redexes.

sequence from t to normal form a descendant of at least one of the redexes in the set is contracted.

Theorem 2.5 (Sekar and Ramakrishnan [15]) *Let $\mathcal{R}$ be an almost orthogonal TRS.*

(1) *Every term that is not a normal form has a necessary set of redexes.*

(2) *Repeated contraction of necessary sets of redexes results in a normal form, whenever the term under consideration has a normal form.*

□

Part (1) holds for any TRS since the set of all redexes in a term is always necessary. For almost orthogonal TRSs the set of all outermost redexes is necessary. In the following we assume that necessary sets of redexes consist of pairwise disjoint redexes. This entails no loss of generality. The proof of part (2) in [15] actually yields the hyper normalization of necessary reduction. Since for orthogonal TRSs singleton sets of needed redexes are necessary, Theorem 2.3(2) is a special case of Theorem 2.5(2).

3 Root-Stability

Definition 3.1 Let $\mathcal{R}$ be a TRS. A term is called *root-stable* if it cannot be rewritten to a redex.

It is not difficult to see that if a term s is not root-stable then $s \xrightarrow{>\varepsilon}{}^* t$ for some redex t. Figure 1 shows the relationship between root-stable terms, redexes, and normal forms. Observe that there are terms that are neither root-stable nor in normal form. We present two closure properties of the set of root-stable terms. The first one, stating that root-stable terms are closed under rewriting, holds for any TRS.

Lemma 3.2 *Let $\mathcal{R}$ be an arbitrary TRS. If s is root-stable and $s \to_{\mathcal{R}}^* t$ then t is root-stable.*

Proof. Obvious. □

For the second one, closure under non-root expansion, we have to restrict ourselves to almost orthogonal TRSs.

Lemma 3.3 *Let $\mathcal{R}$ be an almost orthogonal TRS. If t is root-stable and $s \xrightarrow{>\varepsilon}{}^*_{\mathcal{R}} t$ then s is root-stable.*

Proof. If s is not root-stable then $s \xrightarrow{>\varepsilon}{}^* s'$ for some redex s'. Lemma 2.1 yields a term t' such that $s' \xrightarrow{>\varepsilon}{}^* t'$ and $t \xrightarrow{>\varepsilon}{}^* t'$. Lemma 3.2 shows that t' is root-stable. Since s' is a redex and in almost orthogonal TRSs redexes cannot be destroyed by internal, i.e. non-root, contractions, t' is a redex. But a term cannot be both root-stable and a redex. □

The above lemma does not hold for weakly orthogonal TRSs (which are left-linear TRSs with trivial critical pairs). For instance, the term $f(b)$ is root-stable with respect to the TRS

$$\begin{aligned} f(a) &\to f(b) \\ a &\to b \end{aligned}$$

We have $f(a) \xrightarrow{>\varepsilon} f(b)$ by using the second rewrite rule. The term $f(a)$ is not root-stable as it is a redex.

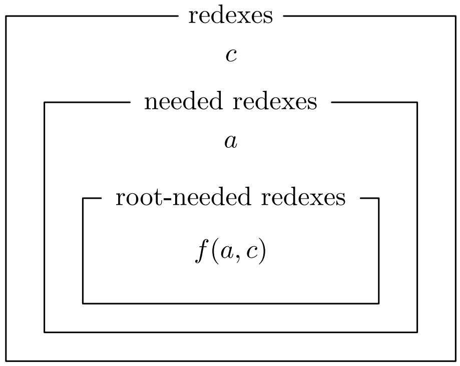
4 Root-Neededness

Definition 4.1 Let $\mathcal{R}$ be a TRS. A redex Δ in a term t is called *root-needed* if in every rewrite sequence from t to a root-stable term a descendant of Δ is contracted.

Clearly every root-needed redex is needed. Figure 2 shows that the inclusion is strict.

In this section we prove that for orthogonal TRSs every non-root-stable term has a root-needed redex. We start with a useful observation.

Lemma 4.2 *Let $\mathcal{R}$ be an orthogonal TRS. If a term rewrites to a redex then the pattern of the first such redex is unique.*



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

Figure 2: Root-needed and needed redexes.

Proof. Suppose $t \xrightarrow{>\varepsilon}{}^* l_1\sigma_1$ and $t \xrightarrow{>\varepsilon}{}^* l_2\sigma_2$ for some rewrite rules $l_1 \to r_1$, $l_2 \to r_2$ and substitutions σ_1, σ_2. We have to show that $l_1 = l_2$. From Lemma 2.1 we obtain a term t' such that $l_1\sigma_1 \xrightarrow{>\varepsilon}{}^* t'$ and $l_2\sigma_2 \xrightarrow{>\varepsilon}{}^* t'$. Since redexes cannot be destroyed by internal contractions, we have $t' = l_1\tau_1 = l_2\tau_2$ for some substitutions τ_1, τ_2. Orthogonality implies that $l_1 = l_2$. $\square$

The above lemma does not hold for almost orthogonal TRSs. Consider for instance the parallel-or TRS of Section 2 and the term $(\top \vee \top) \vee (\top \vee \top)$. If we contract the redex $\top \vee \top$ at position 1, we obtain a redex with respect the first rewrite rule, whereas contraction of the redex at position 2 yields a redex with respect to the second rewrite rule. Note that neither of the two redexes is root-needed.

Theorem 4.3 *Let $\mathcal{R}$ be an orthogonal TRS. Every non-root-stable term has a root-needed redex.*

Proof. Let t be a non-root-stable term. By induction on the structure of t we show that t contains a root-needed redex. If t is a redex then t itself is root-needed because redexes cannot be destroyed by internal contractions. So suppose that t is not a redex. If t does not have a root-stable reduct then all its redexes are root-needed. So suppose that t has a root-stable reduct. First observe that every rewrite sequence from t to root-stable form must contain a root rewrite step, for otherwise t would be root-stable as a consequence of Lemma 3.3. So every rewrite sequence A from t to root-stable form t' can be written as $t \xrightarrow{>\varepsilon}{}^+ \Delta_A \xrightarrow{\varepsilon} t'' \to^* t'$. According to Lemma 4.2 the redex pattern of Δ_A doesn't depend on A. So there is a left-hand side l of a rewrite rule such that every Δ_A is an instance of l. Let P be the set of positions of non-root-stable proper subterms of t. Because t is not an instance of l—recall that t is not a redex—the set $P \cap \mathcal{P}\mathrm{os}_{\mathcal{F}}(l)$ is non-empty. Let p be a minimal (with respect to the prefix order $<$ on positions) element of $P \cap \mathcal{P}\mathrm{os}_{\mathcal{F}}(l)$. As $t_{|p}$ is a proper subterm of t, we can apply the induction hypothesis. This yields a root-needed redex Δ in $t_{|p}$. Let $A\colon t \xrightarrow{>\varepsilon}{}^+ \Delta_A \to^* t'$ be an arbitrary rewrite sequence from t to root-stable form t'. We claim that Δ is contracted in the subsequence $B\colon t \xrightarrow{>\varepsilon}{}^+ \Delta_A$. Due to the minimality of p there are no rewrite steps in B at positions above p. Hence in B the subterm $t_{|p}$ must be rewritten to $(\Delta_A)_{|p}$. The latter term is root-stable as a consequence of orthogonality. Because Δ is root-needed in $t_{|p}$ it must be contracted in B. We conclude that Δ is also root-needed in t. $\square$

Although the above theorem doesn't say anything about root-stable terms that are not in normal form, it is very easy to see that root-needed redexes in maximal non-root-stable subterms of a term t are needed in t. Hence Theorem 4.3 strengthens Theorem 2.3(1). Our proof is much simpler than the one in Huet and Lévy [6].

Definition 4.4 Let $\mathcal{R}$ be a TRS. A set of (pairwise disjoint) redexes in a term t is called *root-necessary* if in every rewrite sequence from t to a root-stable term a descendant of at least one of the redexes in the set is contracted.

For almost orthogonal TRSs the set of all outermost redexes in a non-root-stable term is root-necessary.

5 Root-Normalization

Our next major result states that for almost orthogonal TRSs repeated contraction of root-necessary sets of redexes results in a root-stable term, whenever the latter exists. In particular, root-needed reduction is root-normalizing for orthogonal TRSs. To this end we adapt the elegant proof of Theorem 2.5(2) by Sekar and Ramakrishnan [15].

Throughout this section we assume that we are dealing with almost orthogonal TRSs. We write $s \nparallel t$ if t

can be obtained from s by contracting a set of pairwise disjoint redexes in s. The relation $\mathrel{+\!\!\!\!\!\rightarrow}$ is called *parallel rewriting*. The number of redexes contracted in a parallel rewrite step $A\colon s \mathrel{+\!\!\!\!\!\rightarrow} t$ is denoted by $|A|$. Let $A\colon t_0 \mathrel{+\!\!\!\!\!\rightarrow} t_1 \mathrel{+\!\!\!\!\!\rightarrow} \dots \mathrel{+\!\!\!\!\!\rightarrow} t_n$ be a parallel rewrite sequence. (Parallel rewrite sequences are called *multiderivations* in [6, 15].) We denote the i-th parallel rewrite step $t_{i-1} \mathrel{+\!\!\!\!\!\rightarrow} t_i$ by A_i, so $A = A_1; A_2; \dots; A_n$ where $B; C$ denotes the concatenation of the parallel rewrite sequences B and C. The subsequence $t_{i-1} \mathrel{+\!\!\!\!\!\rightarrow} \dots \mathrel{+\!\!\!\!\!\rightarrow} t_j$ of A is denoted by $A[i,j]$. Let $A\colon s \mathrel{+\!\!\!\!\!\rightarrow} t_1$ and $B\colon s \mathrel{+\!\!\!\!\!\rightarrow} t_2$ be parallel rewrite steps. Let P be the set of positions of the redexes contracted in A. The parallel rewrite step starting from t_2 in which all redexes at positions in $P\backslash B$ are contracted is denoted by $A\backslash B$. This is called the *projection* of A over B. In the following we also project parallel rewrite sequences over parallel rewrite steps. Two parallel rewrite sequences $A\colon s_1 \mathrel{+\!\!\!\!\!\rightarrow}^* t_1$ and $B\colon s_2 \mathrel{+\!\!\!\!\!\rightarrow}^* t_2$ are *permutation equivalent*, denoted by $A \simeq B$, if $s_1 = t_1$, $s_2 = t_2$, and $p\backslash A = p\backslash B$ for all redex positions p in s_1. The following result, known as the Parallel Moves Lemma (cf. Lemma 2.4 in [6]), is illustrated in Figure 3.

Lemma 5.1 *Let $\mathcal{R}$ be an almost orthogonal TRS. If A and B are parallel rewrite steps starting from the same term then $A; (B\backslash A) \simeq B; (A\backslash B)$.* □

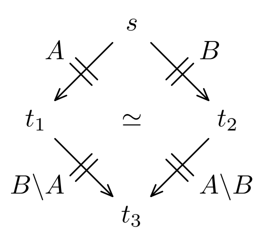


Figure 3: Parallel Moves Lemma.

Definition 5.2 With every parallel rewrite sequence A of length n we associate the n-tuple

$$\text{cost}(A) = (|A_n|, \dots, |A_1|).$$

We define a proper order $>$ on parallel rewrite sequences as follows: $A > B$ if and only if A and B have the same length and $\text{cost}(A) >_{\text{lex}} \text{cost}(B)$, i.e., there exists an $i \in \{1, \dots, n\}$ such that $|A_i| > |B_i|$ and $|A_j| = |B_j|$ for every $i < j \leqslant n$. We write $A \gtrsim B$ if $A > B$ or $\text{cost}(A) = \text{cost}(B)$.

It is easy to see that $\gtrsim$ is a well-founded preorder on parallel rewrite sequences. Note that parallel rewrite sequences of different length are incomparable. (This differs from the order defined in [15].)

Definition 5.3 Let A be a parallel rewrite sequence and B be a parallel rewrite step starting from the same term. We say that B *interferes* with A if a descendant of a redex contracted in B is contracted in A. We write $B \perp A$ if B does not interfere with A. So $B \perp A$ if and only if $\Delta \perp A$ for every redex Δ contracted in B.

The following lemma corresponds to Lemma 8 in [15].

Lemma 5.4 *Let $A\colon s \mathrel{+\!\!\!\!\!\rightarrow}^* s_n$ and $B\colon s \mathrel{+\!\!\!\!\!\rightarrow} t$ be such that $B \perp A$. If s_n is root-stable then there exists a $C\colon t \mathrel{+\!\!\!\!\!\rightarrow}^* t_n$ such that $A \gtrsim C$ and t_n is root-stable. Moreover, if s_n is a normal form, we may assume that $t_n = s_n$.*

Proof. Let us denote a parallel rewrite step in which all contracted redexes are above or disjoint from descendants of redexes contracted in B by $\mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle}$ and a parallel rewrite step in which all contracted redexes are below descendants of redexes contracted in B by $\mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown}$. The proof, which is illustrated in Figure 4 for the case $n = 4$, consists of two steps. In the first step we transform A into a parallel rewrite sequence $A'\colon s \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} s_1^1 \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} \cdots \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} s_n^n$ with $A \gtrsim A'$. This is achieved as follows. Because descendants of redexes contracted in B are not contracted in A, every step A_i in A can be partitioned into two parts: $A_i^1\colon s_{i-1} \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} s_i^1$ and $A_i^2\colon s_i^1 \mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown} s_i$. (By convention $s_0 = s$.) Clearly $|A_i| \geqslant |A_i^1|, |A_i^2|$. It is not difficult to see that the following commutation property holds: $\mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown} \cdot \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} \subseteq \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} \cdot \mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown}$. Moreover, the number of redexes contracted in the two $\mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle}$-steps can be assumed to be the same. Using this property $\frac{1}{2}n(n-1)$ times yields parallel rewrite sequences $A'\colon s \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} s_1^1 \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} \cdots \mathrel{+\!\!\!\!\!\rightarrow}^{\vartriangle} s_n^n$ and $s_n^n\ (\mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown})^*\ s_n$. (We have $A'_1 = A_1^1$.) By construction we have $|A_i^1| = |A'_i|$ and therefore $A \gtrsim A'$. In the second step of the construction we use the Parallel Moves Lemma n times to project A' over B, resulting in the parallel rewrite sequence $C = A'\backslash B\colon t \mathrel{+\!\!\!\!\!\rightarrow} t_1 \mathrel{+\!\!\!\!\!\rightarrow} \cdots \mathrel{+\!\!\!\!\!\rightarrow} t_n$. Because redexes contracted in A' can neither be duplicated nor erased by redexes contracted in B, we clearly have $\text{cost}(A') = \text{cost}(C)$ and thus $A \gtrsim C$. It remains to show that t_n is root-stable. From $s_n^n\ (\mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown})^*\ s_n$ we infer $s_n^n \xrightarrow{>\varepsilon}{}^* s_n$. We also have $s_n^n \mathrel{+\!\!\!\!\!\rightarrow} t_n$. From Lemmata 3.2 and 3.3 we conclude that t_n inherits root-stability from s_n. If s_n is a normal form then it contains no descendants of redexes contracted in B and hence the parallel rewrite sequence $s_n^n\ (\mathrel{+\!\!\!\!\!\rightarrow}^{\triangledown})^*\ s_n$ must be empty. Therefore $s_n^n = s_n$ is a normal form and thus $s_n^n \mathrel{+\!\!\!\!\!\rightarrow} t_n$ implies $s_n^n = t_n$. □

The following lemma corresponds to Theorem 10 in [15].

Lemma 5.5 *Let $A\colon s \mathrel{+\!\!\!\!\!\rightarrow}^* s_n$ and $B\colon s \mathrel{+\!\!\!\!\!\rightarrow} t$ be such that $B \not\perp A$. If s_n is root-stable then there exists a*

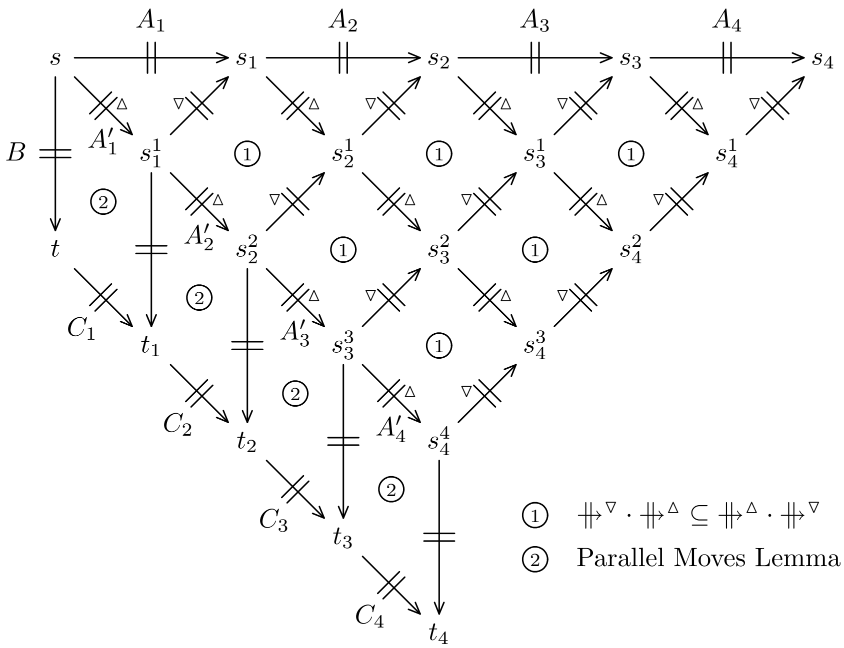


Figure 4: The proof of Lemma 5.4.

$C\colon t \mathrel{+\!\!\!\!\to}^* t_n$ such that $A > C$ and t_n is root-stable. Moreover, if s_n is a normal form, we may assume that $t_n = s_n$.

Proof. The proof is illustrated in Figure 5 for the case $n = 5$. Let A_i be the last step in A where a redex is contracted that descends from a redex contracted in B. (In the figure $i = 3$.) We split A_i into two parts: in $A_i^1\colon s_{i-1} \mathrel{+\!\!\!\!\to} s'_{i-1}$ all descendants of redexes contracted in B are contracted and in $A_i^2\colon s'_{i-1} \mathrel{+\!\!\!\!\to} s_i$ the remaining redexes are contracted. Clearly $|A_i| > |A_i^2|$. Next we apply the Parallel Moves Lemma $i-1$ times to project $A[1, i-1]$ over B. This yields a parallel rewrite sequence $C[1, i-1]\colon t \mathrel{+\!\!\!\!\to}^* t_{i-1}$ and a parallel rewrite step $B'\colon s_{i-1} \mathrel{+\!\!\!\!\to} t_{i-1}$. Let $B'' = B' \backslash A_i^1$. We have $B''\colon s'_{i-1} \mathrel{+\!\!\!\!\to} t_{i-1}$. Now we apply Lemma 5.4 to $A_i^2; A[i+1, n]$ and B'', resulting in a parallel rewrite sequence $C[i, n]\colon t_{i-1} \mathrel{+\!\!\!\!\to}^* t_n$ with $A_i^2; A[i+1, n] \gtrsim C[i, n]$ and t_n root-stable. In particular, $|A_i^2| \geqslant |C_i|$ and thus $|A_i| > |C_i|$. Since $|A_j| \geqslant |C_j|$ for all $i < j \leqslant n$ we conclude that $A > C$. □

The root-normalization of root-necessary reduction is an easy consequence of Lemma 5.5. Using also Lemma 5.4, we obtain the hyper root-normalization of root-necessary reduction.

Theorem 5.6 *Let t be a term that has a root-stable reduct. There are no parallel rewrite sequences starting from t that contain infinitely many root-necessary steps.*

Proof. Let $A\colon t \mathrel{+\!\!\!\!\to}^* t_n$ be a root-normalizing parallel rewrite sequence. We prove the result by induction on A with respect to the well-founded order $>$ on parallel rewrite sequences. If A is a minimal element then $\text{cost}(A) = (0, 0, \ldots, 0)$ and thus $t = t_n$. Hence no reduct of t has root-necessary sets of redexes. For the induction step we reason as follows. Suppose to the contrary that $B = B_1; B_2; \ldots$ is an infinite parallel rewrite sequence starting at t that contains infinitely many root-necessary steps. Let $B_m\colon t^{m-1} \mathrel{+\!\!\!\!\to} t^m$ be the first parallel step in which a root-necessary set of redexes is contracted. According to Lemmata 5.4 and 5.5 there exists a parallel rewrite sequence $A^{m-1}\colon t^{m-1} \mathrel{+\!\!\!\!\to}^* t_n^{m-1}$ with t_n^{m-1} root-stable such that $A \gtrsim A^{m-1}$. By definition of root-neededness, B_m interferes with A^{m-1}. Hence, according to Lemma 5.5, there exists a parallel rewrite sequence $A^m\colon t^m \mathrel{+\!\!\!\!\to}^* t_n^m$ with t_n^m root-stable such that $A^{m-1} > A^m$. So $A > A^m$. According to the induction hypothesis the rewrite sequence $B_{m+1}; B_{m+2}; \ldots$ doesn't contain infinitely many root-necessary steps. This contradicts our assumption. □

Corollary 5.7 *Root-needed reduction is hyper root-normalizing for orthogonal TRSs.* □

Corollary 5.8 *Parallel-outermost is hyper root-normalizing for almost orthogonal TRSs.* □

6 Normalization

In this section we study the relationship between root-normalization and normalization of reduction strategies. We first consider (non-deterministic) sequential strategies.

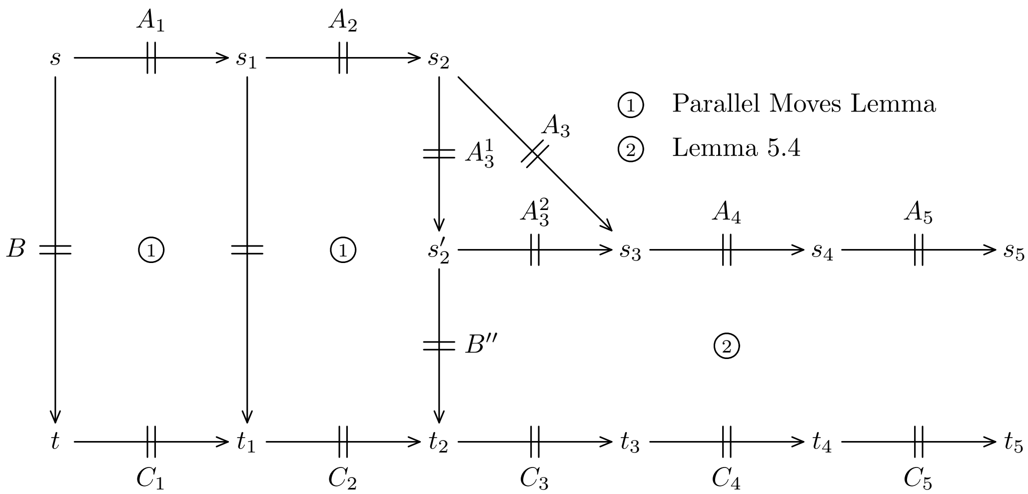


Figure 5: The proof of Lemma 5.5.

Definition 6.1 Let $\mathcal{R}$ be a TRS. A *sequential reduction strategy* for $\mathcal{R}$ is a mapping $\mathcal{S}$ that assigns to every term t not in normal form a non-empty set of redex occurrences in t. We write $t \to^{\mathcal{S}} t'$ if $t \to t'$ by contracting a redex $\Delta \in \mathcal{S}(t)$.

For TRSs that are not almost orthogonal we also need to supply the rewrite rule according to which the selected redex is to be contracted since a redex may have more than one contractum. See [1] for a more precise definition.

According to the above definition needed reduction is a sequential reduction strategy, which assigns to every non-normal form the set of needed redexes occurring in it, for orthogonal TRSs. Root-needed reduction doesn't qualify as a sequential reduction strategy (for orthogonal TRSs) because a root-stable term that is not in normal form doesn't have root-needed redexes. This problem is easily overcome—cf. the discussion following Theorem 4.3—by *defining* root-needed reduction as the sequential reduction strategy $\mathcal{S}$ that is inductively defined as follows:

$$\mathcal{S}(t) = \begin{cases} \{\Delta \mid \Delta \text{ is a root-needed redex in } t\} \\ \qquad \text{if } t \text{ is not root-stable,} \\ \mathcal{S}(t_1) \cup \cdots \cup \mathcal{S}(t_n) \\ \qquad \text{if } t = f(t_1, \ldots, t_n) \text{ is root-stable.} \end{cases}$$

Definition 6.2 A sequential reduction strategy $\mathcal{S}$ for a TRS $\mathcal{R}$ is called *normalizing* if there are no infinite $\mathcal{S}$-rewrite sequences starting from a term that has a normal form. We call $\mathcal{S}$ *root-normalizing* if for all terms t that have a root-stable reduct, every infinite $\mathcal{S}$-rewrite sequence starting from t contains a root-stable term.

Not every normalizing strategy is root-normalizing. Consider for instance needed reduction for the orthogonal TRS

$$\begin{array}{rcl} f(x) & \to & g(x) \\ a & \to & a \end{array}$$

Because the term $f(a)$ doesn't have a normal form, redex a is trivially needed. Hence $f(a) \to f(a) \to f(a) \to \cdots$ is an infinite rewrite sequence in which only needed redexes are contracted. The term $f(a)$ is not root-stable but has a root-stable reduct: $f(a) \to g(a)$.

Conversely, there exist root-normalizing reduction strategies that are not normalizing. Consider for instance the orthogonal TRS

$$\begin{array}{rcl} f(x) & \to & g(a) \\ b & \to & b \end{array}$$

and the strategy $\mathcal{S}$ that always selects the (unique) outermost redex, except when it faces the term $g(f(b))$ in which case the non-outermost redex b is selected. It is easy to see that $\mathcal{S}$ is root-normalizing. However, $\mathcal{S}$ is not normalizing because $g(f(b)) \to^{\mathcal{S}} g(f(b)) \to^{\mathcal{S}} \cdots$ while $g(f(b))$ does have a normal form: $g(f(b)) \to g(g(a))$.

Definition 6.3 A sequential reduction strategy $\mathcal{S}$ is said to be *context-free* if for all root-stable terms $t = f(t_1, \ldots, t_i, \ldots, t_n)$ and $i \in \{1, \ldots, n\}$ such that $t \to^{\mathcal{S}} f(t_1, \ldots, t_i', \ldots, t_n)$ we have $t_i \to^{\mathcal{S}} t_i'$.

The strategy $\mathcal{S}$ defined above is not context-free as $f(b) \to^{\mathcal{S}} f(b)$ doesn't hold. Virtually all strategies found in the literature are context-free. Root-needed reduction defined above is trivially context-free.

Theorem 6.4 *Let $\mathcal{R}$ be a confluent TRS. Every context-free root-normalizing sequential reduction strategy for $\mathcal{R}$ is normalizing.*

Proof. Let $\mathcal{S}$ be a context-free root-normalizing sequential reduction strategy for $\mathcal{R}$. By induction on the structure of the normal form t we show that there are no infinite $\mathcal{S}$-rewrite sequences starting from a term s with $s \to^* t$. If t is a variable or a constant then, due to confluence of $\mathcal{R}$, t is the only root-stable reduct of s. Because $\mathcal{S}$ is root-normalizing, this implies that s doesn't admit infinite $\mathcal{S}$-rewrite sequences. Let $t = f(t_1, \ldots, t_n)$ with $n \geqslant 1$. Suppose to the contrary that $A\colon s \to^{\mathcal{S}} s_1 \to^{\mathcal{S}} s_2 \to^{\mathcal{S}} \cdots$ is infinite. Because $\mathcal{S}$ is root-normalizing and s has a root-stable reduct, viz. t, there exists an $i \geqslant 0$ such that s_i is root-stable. We have $\mathrm{root}(s_i) = f$ due to confluence of $\mathcal{R}$. Hence we may write $s_i = f(u_1, \ldots, u_n)$. Because A is infinite and $\mathcal{S}$ is context-free, there exists an argument u_j of s_i that admits an infinite $\mathcal{S}$-rewrite sequence. Confluence of $\mathcal{R}$ yields $u_j \to^* t_j$. According to the induction hypothesis there are no infinite $\mathcal{S}$-rewrite sequences starting from the term u_j, which is a contradiction. □

Confluence is essential for the validity of Theorem 6.4. Consider for instance the non-confluent TRS

$$\begin{array}{rcl} a & \to & b \\ a & \to & f(c) \\ c & \to & c \end{array}$$

and the reduction strategy that always contracts redex a using the second rewrite rule. Actually, confluence can be weakened to the *normal form property* without invalidating Theorem 6.4. A TRS is said to have the normal form property if all terms that have a normal form are confluent. (This definition is equivalent to the one given in Klop [11].) Note that the above TRS doesn't have the normal form property because the non-confluent term a has a normal form.

The above result easily generalizes to non-sequential reduction strategies. A (non-sequential) reduction strategy for a TRS $\mathcal{R}$ is a mapping $\mathcal{S}$ that assigns to every term t not in normal form a non-empty set of finite non-empty rewrite sequences starting from t. We write $t \to^{\mathcal{S}} t'$ if $t \to^+ t' \in \mathcal{S}(t)$. Definition 6.2 applies also to non-sequential reduction strategies. We modify Definition 6.3 as follows: A reduction strategy $\mathcal{S}$ is context-free if for all root-stable terms $t = f(t_1, \ldots, t_i, \ldots, t_n)$ and $i \in \{1, \ldots, n\}$ such that $t \to^{\mathcal{S}} f(t'_1, \ldots, t'_i, \ldots, t'_n)$ and the subsequence from t_i to t'_i is non-empty we have $t_i \to^{\mathcal{S}} t'_i$. Now the proof of Theorem 6.4 carries verbatim over to non-sequential reduction strategies.

Theorem 6.5 *Let $\mathcal{R}$ be a confluent TRS. Context-free root-normalizing reduction strategy for $\mathcal{R}$ are normalizing.* □

7 Infinitary Normalization

If a term does not have a normal form, all its redexes are needed. As a consequence, needed reduction is not 'infinitary normalizing' (below we formalize this intuitively clear concept): redex a in the term $f(a)$ is needed with respect to the TRS

$$\begin{array}{rcl} f(x) & \to & g(f(x)) \\ a & \to & a \end{array}$$

while $f(a)$ has the 'infinite normal form' $g(g(g(\ldots)))$. Khasidashvili [10] (see also Maranget [12]) proposed a different definition of neededness. He calls a redex Δ in a term t *essential* if Δ has at least one descendant in t' for any term t' such that $A\colon t \to^* t'$ with $\Delta \perp A$. So the only way to eliminate an essential redex is by contraction (of its descendants). Note that for terms having a normal form needed and essential redexes coincide. Khasidashvili writes in [10] that essential redexes "contribute to the finite or infinite result of a computation". However, this is not true as the very same example shows: because the TRS is non-erasing, redex a in $f(a)$ is essential. We already observed that redex a does not contribute to the infinite normal form $g(g(g(\ldots)))$ of $f(a)$. This suggests that root-neededness is more fundamental than neededness (and essentiality) when it comes to infinitary normalization.

Before we can show this formally, we need to get a grip on the concept of infinitary normalization. We are interested in infinite rewrite sequences of length ω whose limit (with respect to the standard notion of Cauchy convergence [2, 9]) is an infinite normal form. Unlike Kennaway *et al.* [9] we don't consider transfinite rewrite sequences whose length exceeds ω since such sequences don't make sense from a practical computational point of view. Consider the infinite orthogonal TRS (based on a similar example in [9])

$$\begin{array}{rcll} f(g^i(h(x)) & \to & f(g^{i+1}(h(x))) & \text{for all } i \geqslant 0 \\ a & \to & a & \end{array}$$

The limit of the infinite rewrite sequence $t = f(h(a)) \to f(g(h(a))) \to f(g(g(h(a)))) \to \cdots$ is the infinite normal form $f(g(g(g(\ldots))))$. Since t doesn't have a root-stable reduct, all its redexes are root-needed but repeatedly contracting redex a doesn't yield the desired result. The problem with this infinite normalizing sequence is that all steps take place at the root position hence no function symbol in its limit is produced after finitely many steps. For this reason Kennaway *et al.* develop in [9] the theory of transfinite rewriting for *strongly converging* rewrite sequences. In a strongly converging infinite rewrite sequence the depth of the contracted redexes tends to infinity. More formally, an infinite rewrite sequence $t_0 \to t_1 \to \cdots$ is strongly converging if for all $d \geqslant 0$ there exists an index $i \geqslant 0$ such that the depth of every redex contracted in $t_i \to t_{i+1} \to \cdots$ is at least d. (The depth of a redex in a term is the length of the position at which it occurs.) Also all finite rewrite

(1)	$f(a) \to f(b) \to c$	
(2)	$f(a) \to f(b) \to f(a)$	
(3)	$f(a) \to f(b) \to f(a) \to f(b) \to \cdots$	
(4)	$g(c) \to h(c, g(c)) \to h(c, h(c, g(c))) \to h(c, h(c, h(c, g(c)))) \to \cdots$	
(5)	$g(a) \to h(a, g(a)) \to h(a, h(a, g(a))) \to h(a, h(a, h(a, g(a)))) \to \cdots$	

TRS

$$\begin{array}{rcl} a & \to & b \\ b & \to & a \\ f(x) & \to & c \\ g(x) & \to & h(x, g(x)) \end{array}$$

Figure 6: Classification of rewrite sequences.

sequences are strongly converging. Note that every infinite strongly converging rewrite sequence has a limit. Let $t_0 \to t_1 \to \cdots$ be an infinite strongly converging rewrite sequence. If its limit t_ω is a (necessarily infinite) normal form then for all $d \geqslant 0$ there exists an index $i \geqslant 0$ such that the depth of every non-root-stable term in t_i is at least d. So function symbols in t_ω are produced after finitely many steps. In [9, Theorem 9.1] it is remarked that orthogonal TRSs such that the depth of the left-hand sides of the rewrite rules is bounded have the property that for every infinite rewrite sequence whose limit is a normal form there exists a strongly converging rewrite sequence whose limit is the same normal form. Hence in particular for finite TRSs the restriction to strongly converging rewrite sequences entails no loss of generality.

For a proper understanding of the following definition, it is helpful to distinguish rewrite sequences into five different categories: (1) finite rewrite sequences ending in a normal form, (2) finite rewrite sequences not ending in a normal form, (3) infinite rewrite sequences that are not strongly converging, (4) infinite strongly converging rewrite sequences whose limit is a normal form, and (5) infinite strongly converging rewrite sequences whose limit is not a normal form. See Figure 6 for an example.

Definition 7.1 A rewrite sequence is called *infinitary normalizing* if it strongly converges to a (possibly infinite) normal form. An infinite rewrite sequence that is not infinitary normalizing is called *perpetual*. (Our notion of perpetual rewrite sequence is different from [9, Definition 8.8].) A reduction strategy $\mathcal{S}$ for a TRS is called *infinitary normalizing* if there are no perpetual $\mathcal{S}$-rewrite sequences starting from terms that admit an infinitary normalizing rewrite sequence.

According to the above classification of rewrite sequences, categories (1) and (4) consist of all infinitary normalizing rewrite sequences and categories (3) and (5) consist of all perpetual rewrite sequences.

Consider the leftmost-outermost reduction strategy applied to the term a with respect to the TRS

$$a \quad \to \quad f(a, a)$$

The resulting perpetual rewrite sequence $a \to f(a, a) \to f(f(a, a), a) \to f(f(f(a, a), a), a) \to \cdots$ shows that we cannot hope to achieve infinitary normalization without imposing a fairness condition.

Definition 7.2 An infinite rewrite sequence $t_0 \to t_1 \to \cdots$ is called *fair* if for every $i \geqslant 0$ and every maximal non-root-stable subterm s of t_i that has a root-stable reduct there is a $j \geqslant i$ such that the position of the redex contracted in the step $t_j \to t_{j+1}$ is below the position of s in t_i.

So in a fair sequence maximal non-root-stable subterms are never left undisturbed. Our notion of fairness, although much weaker than the one in Kennaway *et al.* [9, Definition 8.3], is sufficient for infinitary normalization. (The fairness notion in [9] can be shown to be equivalent to O'Donnell's outermost-fairness [1, 13].)

Definition 7.3 A reduction strategy $\mathcal{S}$ for a TRS is called *infinitary fair-normalizing* if there are no perpetual fair $\mathcal{S}$-rewrite sequences starting from terms that admit an infinitary normalizing rewrite sequence.

Theorem 7.4 *Let $\mathcal{R}$ be a confluent TRS. Context-free root-normalizing reduction strategies for $\mathcal{R}$ are infinitary fair-normalizing.*

Proof. Because of the special structure of infinitary normalizing rewrite sequences, the proof of Theorem 6.4 generalizes to the present setting. □

The example at the beginning of this section shows that this theorem does not hold for normalizing reduction strategies. Without the fairness requirement, context-free root-normalizing reduction strategies will produce an infinite part of infinite normal forms.

Theorem 7.5 *Let $\mathcal{S}$ be a context-free root-normalizing reduction strategy for a confluent TRS. The limit of every infinite strongly converging $\mathcal{S}$-rewrite sequence has an infinite stable part.* □

Since every infinite rewrite sequence produced by the parallel-outermost strategy is trivially fair, we obtain the following corollary from Theorem 7.4 and Corollary 5.8.

Corollary 7.6 *Parallel-outermost is infinitary normalizing for almost orthogonal TRSs.* □

It is not difficult to generalize the above results to infinitary hyper normalization. Another, less trivial, consequence of Theorem 7.4 is the infinitary normalization of the sequential reduction strategy $\mathcal{S}_\omega$ of Antoy and Middeldorp [1] for almost orthogonal TRSs.

Corollary 7.7 *$\mathcal{S}_\omega$ is infinitary normalizing for almost orthogonal TRSs.*

Proof (sketch). The context-freeness of $\mathcal{S}_\omega$ is a direct consequence of Lemma 6.4 in [1]. From the proof of Theorem 6.7 in [1] it easily follows that if $\mathcal{S}_\omega$ fails to root-normalize some term t then there exists a rewrite sequence starting from t which is outermost-fair but not root-normalizing. The root-normalization of outermost-fair reduction for almost orthogonal TRSs can be shown with help of (the proofs of) Lemmata 5.4 and 5.5. Therefore $\mathcal{S}_\omega$ is root-normalizing for the same class of TRSs. Finally, fairness of the infinite rewrite sequences produced by $\mathcal{S}_\omega$ is an easy consequence of Lemma 6.3 in [1], using the facts that (1) if a non-root-stable term has a root-stable reduct then it cannot be $\mathcal{S}_\omega$-cyclic and (2) a term is $\mathcal{S}_\omega$-cyclic if and only if all its maximal root-stable subterms are $\mathcal{S}_\omega$-cyclic. Part (2) can be proved by induction on the term structure. □

Let us conclude this section by comparing our results with the ones of Kennaway *et al.* [9]. Although their setting (TRSs with possibly infinite right-hand sides, strongly converging rewrite sequences of 'arbitrary' length) is more general, our restricted setting is quite sufficient from a programming point of view. Kennaway *et al.* redefine needed redexes with respect to strongly converging rewrite sequences and extend Theorem 2.3 accordingly. We believe that our (much simpler) concept of root-needed redexes is the right generalization of Huet and Lévy's needed redexes when dealing with infinitary normalization (as well as normalization as argued in the preceding sections). Observe that the first example in the previous section shows that needed reduction as defined in [9] is not root-normalizing. As a final remark we mention that the results on transfinite normalization obtained in [9] are restricted to *orthogonal* TRSs.

8 Conclusion

In this paper we investigated call by need computations to root-stable form for orthogonal and almost orthogonal TRSs. For orthogonal TRSs it is in general undecidable whether a redex is root-needed. Also, root-stability is undecidable. For instance, it is straightforward to associate with every (deterministic) Turing machine M an orthogonal TRS $\mathcal{R}_M$ such that $A \to^* B$ if and only if M halts on empty tape. Here A and B are constants with B in normal form. For the orthogonal TRS $\mathcal{R}_M \cup \{f(B) \to B\}$ the statements $A \to^* B$, $f(A)$ is not root-stable, redex A in $f(A)$ is root-needed, and $f(A)$ has root-needed redexes are equivalent.

In a forthcoming joint paper with Irène Durand we present a unifying approach to decidable approximations of root-stability, root-neededness, neededness, and call by need, which simplifies and generalizes many works in this direction (e.g. [4, 6, 7, 8, 14]).

Acknowledgements

I thank Irène Durand, Vincent van Oostrom, and Fer-Jan de Vries for helpful discussions, Nachum Dershowitz for suggesting the terminology context-free (in Definition 6.3), and Femke van Raamsdonk for carefully reading the paper. The work reported in this paper is partially supported by the Advanced Information Technology Program (AITP) of the Information Technology Promotion Agency (IPA).

References

[1] S. Antoy and A. Middeldorp, *A Sequential Reduction Strategy*, Theoretical Computer Science **165**(1), pp. 75–95, 1996.

[2] A. Arnold and M. Nivat. *The Metric Space of Infinite Trees. Algebraic and Topological Properties*, Fundamenta Informaticae **4**, pp. 445–476, 1980.

[3] H.P. Barendregt, J.R. Kennaway, J.W. Klop, and M.R. Sleep, *Needed Reduction and Spine Strategies for the Lambda Calculus*, Information and Computation **75**(3), pp. 191–231, 1987.

[4] H. Comon, *Sequentiality, Second-Order Monadic Logic and Tree Automata*, Proceedings of the 10th IEEE Annual Symposium on Logic in Computer Science, San Diego, pp. 508–517, 1995.

[5] N. Dershowitz and J.-P. Jouannaud, *Rewrite Systems*, in: Handbook of Theoretical Computer Science, Vol. B (ed. J. van Leeuwen), North-Holland, pp. 243–320, 1990.

[6] G. Huet and J.-J. Lévy, *Computations in Orthogonal Rewriting Systems, I and II*, in: Computational Logic, Essays in Honor of Alan Robinson (eds. J.-L. Lassez and G. Plotkin), The MIT Press, pp. 396–443, 1991. Previous version: *Call by Need Computations in Non-Ambiguous Linear Term Rewriting Systems*, report 359, INRIA, 1979.

[7] F. Jacquemard, *Decidable Approximations of Term Rewriting Systems*, Proceedings of the 7th International Conference on Rewriting Techniques and Applications, New Brunswick, Lecture Notes in Computer Science **1103**, pp. 362–376, 1996.

[8] J.R. Kennaway, *A Conflict Between Call-by-Need Computation and Parallelism*, Proceedings of the 4th Workshop on Conditional Term Rewriting Systems, Jerusalem, Lecture Notes in Computer Science **968**, pp. 247–261, 1995.

[9] J.R. Kennaway, J.W. Klop, M.R. Sleep, and F.-J. de Vries, *Transfinite Reductions in Orthogonal Term Rewriting Systems*, Information and Computation **119**(1), pp. 18–38, 1995.

[10] Z. Khasidashvili, *Optimal Normalization in Orthogonal Term Rewriting Systems*, Proceedings of the 5th International Conference on Rewriting Techniques and Applications, Montreal, Lecture Notes in Computer Science **690**, pp. 243–258, 1993.

[11] J.W. Klop, *Term Rewriting Systems*, in: Handbook of Logic in Computer Science, Vol. 2 (eds. S. Abramsky, D. Gabbay and T. Maibaum), Oxford University Press, pp. 1–116, 1992.

[12] L. Maranget, *La Stratégie Paresseuse*, Thèse de Doctorat, Université de Paris 7, 1992 (in French).

[13] M.J. O'Donnell, *Computing in Systems Described by Equations*, Lecture Notes in Computer Science **58**, 1977.

[14] M. Oyamaguchi, *NV-Sequentiality: A Decidable Condition for Call-by-Need Computations in Term Rewriting Systems*, SIAM Journal on Computation **22**(1), pp. 114–135, 1993.

[15] R.C. Sekar and I.V. Ramakrishnan, *Programming in Equational Logic: Beyond Strong Sequentiality*, Information and Computation **104**(1), pp. 78–109, 1993.