Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science 72 (2007) 87–100
Electronic Notes in Theoretical Computer Science
www.elsevier.com/locate/entcs

Conditional Term Graph Rewriting with Indirect Sharing

Enno Ohlebusch[1]

Faculty of Technology
University of Bielefeld
33501 Bielefeld, Germany

Abstract

For reasons of efficiency, term rewriting is usually implemented by term graph rewriting. In term rewriting, expressions are represented as terms, whereas in term graph rewriting these are represented as directed graphs. Unlike terms, graphs allow a sharing of common subexpressions. In previous work, we have shown that conditional term graph rewriting is a sound and complete implementation for a certain class of CTRSs with strict equality, provided that a minimal structure sharing scheme is used. In this paper, we will show that this is also true for two different extensions of normal CTRSs. In contrast to the previous work, however, a non-minimal structure sharing scheme can be used. That is, the amount of sharing is increased.

Keywords: Conditional term rewriting system, term graph rewriting, structure sharing

1 Introduction

The main result in [15] states that conditional term graph rewriting is a sound and complete implementation of conditional term rewriting for a certain class of CTRSs with strict equality (in a CTRS with strict equality the condition $s_i = t_i$ is satisfied for a substitution σ if $s_i\sigma$ and $t_i\sigma$ are reducible to a common ground constructor term; cf. [9]), provided that a minimal structure sharing scheme is used. Soundness ensures that the graph implementation cannot give incorrect results, while completeness ensures that it gives all results. The proof of this result depends on the validity of the parallel moves lemma for these CTRSs. In this paper, we will show that two different extensions of normal CTRSs also satisfy the parallel moves lemma. As a consequence, it will be proven that conditional term graph rewriting is a sound and complete implementation of the two classes of CTRSs, too. However, in contrast to [15], we do not use a minimal structure sharing scheme here. In other words, the amount of sharing is increased.

[1] Email: `enno@TechFak.Uni-Bielefeld.DE`

doi:10.1016/j.entcs.2002.09.008

type	requirement
1	$\mathcal{V}ar(r) \cup \mathcal{V}ar(c) \subseteq \mathcal{V}ar(l)$
2	$\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$
3	$\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l) \cup \mathcal{V}ar(c)$
4	no restrictions

Table 1

It is well-known that orthogonal normal 2-CTRSs satisfy the parallel moves lemma. Since these systems are the starting point of our investigations, we will first recall the relevant definitions; more details can be found in [16].

In a CTRS $(\mathcal{F}, \mathcal{R})$ rules have the form $l \to r \Leftarrow s_1 = t_1, \ldots, s_k = t_k$ with $l, r, s_1, \ldots, s_k,\ t_1, \ldots, t_k \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. The left-hand side l may not be a variable. We frequently abbreviate the conditional part of the rule, that is, the sequence $s_1 = t_1, \ldots, s_k = t_k$, by c. If a rewrite rule has no conditions, we write $l \to r$, demand that $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$, and call $l \to r$ an unconditional rule. As in [13], rewrite rules $l \to r \Leftarrow c$ will be classified according to the distribution of variables among l, r, and c, as shown in Table 1.

An n-CTRS contains only rewrite rules of type n. For every rule $l \to r \Leftarrow c$, we define the set of *extra variables* by

$$\mathcal{EV}ar(l \to r \Leftarrow c) = (\mathcal{V}ar(r) \cup \mathcal{V}ar(c)) \setminus \mathcal{V}ar(l).$$

Thus a 1-CTRS has no extra variables, a 2-CTRS has no extra variables on the right-hand sides of rules, and a 3-CTRS may contain extra variables on the right-hand sides of rules provided that these also occur in the conditions.

The unconditional TRS obtained from a CTRS $\mathcal{R}$ by omitting the conditions in its rewrite rules is denoted by $\mathcal{R}_u$. Note that $(\mathcal{F}, \mathcal{R}_u)$ is an unconditional TRS in the usual sense provided that $(\mathcal{F}, \mathcal{R})$ is a 2-CTRS. This is not true for 3-CTRSs because rules of type 3 may contain variables on the right-hand sides of rules which do not occur on the corresponding left-hand side. For a CTRS $\mathcal{R}$, notions like left-linearity and constructor term are defined via the system $\mathcal{R}_u$. Since these properties solely depend on the left-hand sides of the system $\mathcal{R}_u$, they are well-defined even if $\mathcal{R}_u$ is not a TRS in the usual sense.

The $=$ symbol in the conditions can be interpreted in different ways which lead to different rewrite relations associated with $\mathcal{R}$. In a *normal* CTRS every rule $l \to r \Leftarrow s_1 = t_1, \ldots, s_k = t_k$ in $\mathcal{R}$ satisfies the restriction that every t_i is a ground normal form with respect to $\mathcal{R}_u$.[2] The rewrite relation $\to_{\mathcal{R}}$ associated with a normal CTRS $\mathcal{R}$ is defined by $\to_{\mathcal{R}} = \bigcup_{n \geq 0} \to_{\mathcal{R}_n}$, where $\to_{\mathcal{R}_0} = \emptyset$ and, for $n > 0$, $s \to_{\mathcal{R}_n} t$ if and only if there is a rule $l \to r \Leftarrow c$ in $\mathcal{R}$, a substitution $\sigma : \mathcal{V} \to \mathcal{T}(\mathcal{F}, \mathcal{V})$ with $\mathcal{D}(\sigma) = \mathcal{V}ar(l \to r \Leftarrow c)$, and a context $C[\ ]$ such that $s = C[l\sigma]$, $t = C[r\sigma]$, and $s_i\sigma \to^*_{\mathcal{R}_{n-1}} t_i$ for all $s_i = t_i$ in c. In other words, in a normal CTRS a condition

[2] The question of whether a term is a normal form with respect to $\mathcal{R}$ is in general undecidable.

$s_i = t_i$ is satisfied for a substitution σ if $s_i\sigma$ reduces to the ground $\mathcal{R}_u$-normal form t_i.

If one drops the ground $\mathcal{R}_u$-normal form requirement in the definition of normal CTRSs and allows variables on the right-hand sides t_i of the conditions, then there are at least two possibilities to define the rewrite relation:

(i) A condition $s_i = t_i$ is satisfied for a substitution σ if $s_i\sigma \to^*_{\mathcal{R}_{n-1}} t_i\sigma$ (oriented systems).

(ii) A condition $s_i = t_i$ is satisfied for a substitution σ if $s_i\sigma$ reduces to a ground $\mathcal{R}_u$-normal form v_i such that $v_i = t_i\sigma$ (oriented systems with normalizing conditions).

Note that both oriented systems and oriented systems with normalizing conditions subsume normal systems. Oriented conditional rewrite systems were studied in [5,7,6,8,3,20,12,16] among others. In particular, Suzuki et al. [20] investigated the class of orthogonal properly oriented right-stable 3-CTRSs (the notions right-stability and proper orientation are defined in Definitions 2.5 and 2.9, respectively). They showed that these systems are level-confluent (a CTRS $\mathcal{R}$ is called *level-confluent* if every relation $\to_{\mathcal{R}_n}$ is confluent) although they do not satisfy the parallel moves lemma. Conditional rules from an oriented system will be denoted by $l \to r \Leftarrow s_1 \to t_1, \ldots, s_k \to t_k$. Oriented systems with normalizing conditions have not yet been studied. They have the advantage over oriented systems that the reduction phase and the matching phase are separated (first try to reduce $s_i\sigma$ to ground $\mathcal{R}_u$-normal form and then check whether t_i matches the normal form). Conditional rules from oriented systems with normalizing conditions will be denoted by $l \to r \Leftarrow s_1 \to^! t_1, \ldots, s_k \to^! t_k$.

Let $l_1 \to r_1 \Leftarrow c_1$ and $l_2 \to r_2 \Leftarrow c_2$ be renamed versions of rewrite rules of $\mathcal{R}$ such that they have no variables in common. Suppose $l_1 = C[t]$ with $t \notin \mathcal{V}$ such that $t\sigma = l_2\sigma$ for a most general unifier σ. We call $\langle C[r_2]\sigma, r_1\sigma\rangle \Leftarrow c_1\sigma, c_2\sigma$ a *conditional critical pair* of $\mathcal{R}$. If the two rules are renamed versions of the same rewrite rule of $\mathcal{R}$, we do not consider the case $C[\,] = \Box$. A conditional critical pair $\langle s, t\rangle \Leftarrow s_1 \to t_1, \ldots, s_n \to t_n$ is *feasible* if there is a substitution σ which satisfies $s_j\sigma \to^*_{\mathcal{R}} t_j\sigma$ for all $j \in \{1, \ldots, n\}$. Otherwise it is called *infeasible*. A conditional critical pair $\langle t, t\rangle \Leftarrow s_1 \to t_1, \ldots, s_n \to t_n$ is called *trivial*.

A left-linear CTRS is *orthogonal* if it has no conditional critical pair, and it is *almost orthogonal* if every conditional critical pair is trivial and results from an overlay (i.e., an overlap at root positions).

It will be shown in this paper that the parallel moves lemma holds for the following classes of CTRSs:

(i) orthogonal right-stable oriented 2-CTRSs,

(ii) orthogonal oriented 3-CTRSs with normalizing conditions that are right-stable and properly oriented.

As a consequence, it will be proven that conditional term graph rewriting is a sound and complete implementation of the two classes of CTRSs.

2 The Parallel Moves Lemma

We start with the parallel moves lemma for orthogonal normal 2-CTRS; see Bergstra and Klop [5].

Definition 2.1 The set $\mathcal{P}os(t)$ of *positions* in $t \in \mathcal{T}(\mathcal{F},\mathcal{V})$ is defined by:

$$\mathcal{P}os(t) = \begin{cases} \{\varepsilon\} & \text{if} \quad t \in \mathcal{V} \\ \{\varepsilon\} \cup \{i.p \mid p \in \mathcal{P}os(t_i) \text{ and } 1 \leq i \leq n\} & \text{if} \quad t = f(t_1,\dots,t_n) \end{cases}$$

A position p within a term t is thus a sequence of natural numbers (where ε denotes the empty sequence and the numbers are separated by dots) describing the path from the root of t to the root of the *subterm occurrence* $t|_p$, where

$$t|_p = \begin{cases} t & \text{if} \quad p = \varepsilon \\ t_i|_q & \text{if} \quad p = i.q,\ 1 \leq i \leq n, \text{ and } \ t = f(t_1,\dots,t_n). \end{cases}$$

Furthermore, we define $\mathcal{VP}os(t) = \{p \in \mathcal{P}os(t) \mid t|_p \in \mathcal{V}\}$. That is, $\mathcal{VP}os(t)$ denotes the set of all *variable positions* in t.

Positions are partially ordered by the so-called *prefix ordering* $\geq$, i.e., $p \geq q$ if there is an o such that $p = q.o$. In this case we say that p is *below* q or q is *above* p. Moreover, if $p \neq q$, then we say that p is *strictly below* q or q is *strictly above* p. Two positions p and q are *independent* or *disjoint*, denoted by $p \parallel q$, if neither one is below the other.

Definition 2.2 Let $A : s \rightarrow_{p,l\rightarrow r\Leftarrow c} t$ be a rewrite step in a CTRS $\mathcal{R}$ and let $q \in \mathcal{P}os(s)$. The set $q\backslash A$ of *descendants* of q in t is defined by:

$$q\backslash A = \begin{cases} \{q\} & \text{if } q < p \text{ or } q \parallel p, \\ \{p.p_3.p_2 \mid r_{|p_3} = l_{|p_1}\} & \text{if } q = p.p_1.p_2 \text{ with } p_1 \in \mathcal{VP}os(l) \\ \emptyset & \text{otherwise.} \end{cases}$$

If $Q \subseteq \mathcal{P}os(s)$, then $Q\backslash A$ denotes the set $\bigcup_{q\in Q} q\backslash A$. The notion of descendant is extended to rewrite sequences in the obvious way.

Definition 2.3 Let $\mathcal{R}$ be a CTRS. We write $s \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} t$ if t can be obtained from s by contracting a set of pairwise disjoint redexes in s by $\rightarrow_{\mathcal{R}_n}$. We write $s \mathrel{+\!\!\!\!\!\!\!\rightarrow} t$ if $s \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} t$ for some $n \in \mathbb{N}$. The minimum such n is called the *depth* of $s \mathrel{+\!\!\!\!\!\!\!\rightarrow} t$. The relation $\mathrel{+\!\!\!\!\!\!\!\rightarrow}$ is called *parallel rewriting*.

The following parallel moves lemma holds for orthogonal normal 2-CTRS; see Bergstra and Klop [5].

Lemma 2.4 *If $u \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_m} u_1$ and $u \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} u_2$, then there is a term u_3 such that $u_1 \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} u_3$ and $u_2 \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_m} u_3$. Moreover, the redexes contracted in the parallel reduction step $u_1 \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} u_3$ ($u_2 \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_m} u_3$) are the descendants in u_1 (u_2) of the redexes contracted in $u \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_n} u_2$ ($u \mathrel{+\!\!\!\!\!\!\!\rightarrow}_{\mathcal{R}_m} u_1$).*

The preceding parallel moves lemma can be extended to almost orthogonal systems and one may allow infeasible conditional critical pairs. It is thus an immediate corollary to Lemma 2.4 that these systems are level-confluent.

Our next goal is to show that orthogonal right-stable oriented 2-CTRS also satisfy the parallel moves lemma.

Definition 2.5 A CTRS $\mathcal{R}$ is called *right-stable* if every rewrite rule $l \to r \Leftarrow s_1 = t_1, \ldots, s_k = t_k$ in $\mathcal{R}$ satisfies the following conditions for all $i \in \{1, \ldots, k\}$:

$$(\mathcal{V}ar(l) \cup \bigcup_{j=1}^{i-1} \mathcal{V}ar(s_j = t_j) \cup \mathcal{V}ar(s_i)) \cap \mathcal{V}ar(t_i) = \emptyset$$

and t_i is either a linear constructor term or a ground $\mathcal{R}_u$-normal form.

Example 2.6 Given a function mod such that evaluating $mod(x, n)$ gives the remainder r of x divided by n (where x and n are natural numbers) and a function eq which tests equality of two natural numbers, we define a function $filter$ which filters all elements out of a list of natural numbers that have remainder r when divided by n.

$$\begin{aligned}
filter(n, r, nil) &\to nil \\
filter(n, r, x : xs) &\to x : filter(n, r, xs) \Leftarrow mod(x, n) \to r', eq(r, r') \to true \\
filter(n, r, x : xs) &\to filter(n, r, xs) \qquad \Leftarrow mod(x, n) \to r', eq(r, r') \to false
\end{aligned}$$

The above rules constitute a left-linear right-stable oriented 2-CTRS.

It is convenient to further partition the extra variables in a right-stable system as follows.

Definition 2.7 The set of all *bound* extra variables in a right-stable conditional rewrite rule $\rho : l \to r \Leftarrow s_1 = t_1, \ldots, s_k = t_k$ from $\mathcal{R}$ is defined by

$$\mathcal{EV}ar^b(\rho) = \bigcup_{i=1}^{k} \mathcal{V}ar(t_i)$$

and the set of *free* variables is $\mathcal{EV}ar^f(\rho) = \mathcal{EV}ar(\rho) \setminus \mathcal{EV}ar^b(\rho)$.

For example, if ρ is the rule

$$f(x) \to true \Leftarrow y * y \to z, mod(x, z) \to s(0)$$

we have $\mathcal{EV}ar^f(\rho) = \{y\}$ and $\mathcal{EV}ar^b(\rho) = \{z\}$.

Lemma 2.8 *The parallel moves lemma holds for orthogonal right-stable oriented 2-CTRSs.*

The restrictions imposed in Lemma 2.8 are really necessary as counterexamples in [5,20] show. Analogously to Lemma 2.4, the preceding parallel moves lemma can be extended to almost orthogonal systems and one may allow infeasible conditional critical pairs. Hence it follows that the CTRS from Example 2.6 is level-confluent.

Next we show that under certain conditions, the parallel moves lemma also holds for oriented 3-CTRSs with normalizing conditions (recall that in these systems a condition $s_i = t_i$ is satisfied for a substitution σ if $s_i\sigma$ reduces to a ground $\mathcal{R}_u$-normal form v_i such that $v_i = t_i\sigma$).

Definition 2.9 A CTRS $\mathcal{R}$ is called *properly oriented* if every conditional rewrite rule $l \to r \Leftarrow s_1 = t_1, \ldots, s_k = t_k$ in $\mathcal{R}$ with $\mathcal{V}ar(r) \not\subseteq \mathcal{V}ar(l)$ satisfies $\mathcal{V}ar(s_i) \subseteq \mathcal{V}ar(l) \cup \bigcup_{j=1}^{i-1} \mathcal{V}ar(t_j)$ for all $1 \leq i \leq k$.

Definition 2.10 An orthogonal oriented 3-CTRS $\mathcal{R}$ with normalizing conditions is called *good-natured* if it is right-stable and properly oriented.

As an example consider the good-natured CTRS $\mathcal{R}_{fib}$ which computes the Fibonacci numbers.

$$\begin{aligned} 0 + x &\to x \\ s(x) + y &\to s(x + y) \\ fib(0) &\to \langle 0, s(0) \rangle \\ fib(s(x)) &\to \langle z, y + z \rangle \Leftarrow fib(x) \to^! \langle y, z \rangle \end{aligned}$$

The following lemma states that in a good-natured CTRS the extra-variables on right-hand sides of rules are not really extra because they belong to the bound variables.

Lemma 2.11 *For every rule $\rho : l \to r \Leftarrow s_1 \to^! t_1, \ldots, s_k \to^! t_k$ in a good-natured CTRS $\mathcal{R}$, we have*

(i) $\mathcal{V}ar(r) \not\subseteq \mathcal{V}ar(l)$ *implies* $\mathcal{EV}ar(\rho) = \mathcal{EV}ar^b(\rho)$,

(ii) $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l) \cup \mathcal{EV}ar^b(\rho)$.

In good-natured CTRSs extra variables which occur on the right-hand side of a rule are bound (get a value) during the evaluation of the conditions. More precisely, if the left-hand side l of a rewrite rule matches a subterm of the term to be evaluated via substitution σ, then all variables in l are bound. Now the conditions are evaluated from left-to-right. The left-hand side s_1 of the first condition always contains only variables which also occur in l and thus already have a binding. Then $s_1\sigma$ is rewritten until a ground $\mathcal{R}_u$-normal form, say t'_1, is obtained. The first condition is satisfied if its right-hand side t_1 matches this normal form t'_1. Now if $t_1 = C'[x_1, \ldots, x_n]$, where all variables are displayed, then t_1 matches t'_1 if and only if $t'_1 = C'[w_1, \ldots, w_n]$. This is because t_1 is linear and its variables do not occur in l (and s_1). Hence every variable in t_1 is bound during the match. Now the left-hand side s_2 of the second condition contains only variables which occurred already left to it (in l and t_1) and are thus bound. The instantiated term s_2 is then reduced to a ground $\mathcal{R}_u$-normal form and so on. If all the conditions are satisfied, then all variables in the conditions are bound in the process of evaluating the conditions. Hence the reduct of $l\sigma$ is well-defined (and unique as we shall see) because the right-hand side r of the rule contains only variables which also appear in the conditions or in

l. These systems allow, for example, local definitions like let and where constructs in good-natured programming.

Lemma 2.12 *Good-natured 3-CTRS also satisfy the parallel moves lemma.*

It is an immediate consequence of the preceding lemma that every good-natured 3-CTRS $\mathcal{R}$ is level-confluent. Again, this parallel moves lemma (and hence level-confluence) can be extended to almost orthogonal systems which may contain infeasible conditional critical pairs. As a matter of fact, by carefully checking the proofs in Suzuki et al. [20], one finds that level-confluence of good-natured 3-CTRS can be proven in the same manner. However, the results of the next section crucially depend on the validity of the parallel moves lemma. This lemma does not hold if we consider oriented systems instead of oriented systems with normalizing conditions as the following simple example taken from [20, Example 4.4] shows.

Example 2.13 Let

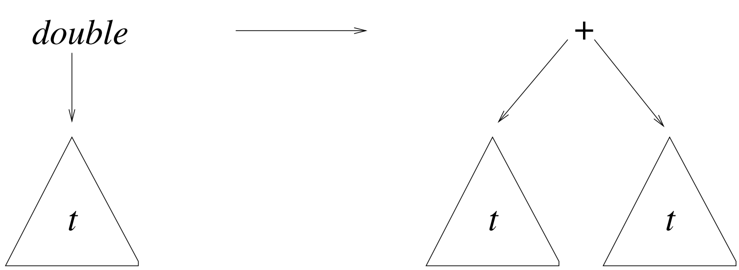
$$\mathcal{R} = \begin{cases} f(x) \to y \Leftarrow x \to y \\ a \to b \\ b \to c \end{cases}$$

Then $f(a) \mathrel{+\!\!\!\!\!\rightarrow}_{\mathcal{R}_1} a$ and $f(a) \mathrel{+\!\!\!\!\!\rightarrow}_{\mathcal{R}_2} c$ but not $a \mathrel{+\!\!\!\!\!\rightarrow}_{\mathcal{R}_2} c$.

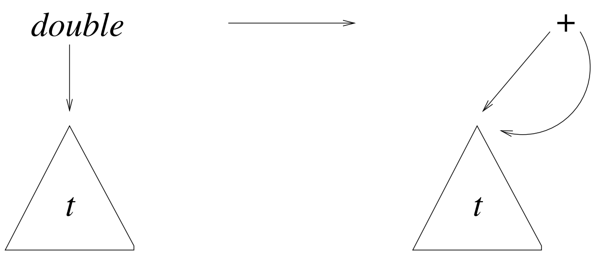
3 Conditional Term Graph Rewriting

For reasons of efficiency, term rewriting is usually implemented by term graph rewriting. In term rewriting, expressions are represented as terms, whereas in term graph rewriting these are represented as directed graphs. Unlike terms, graphs allow a sharing of common subexpressions. In term graph rewriting expressions are evaluated by rule-based graph transformations. Here we will only consider directed *acyclic* graphs.

In order to illustrate the advantage of term graph rewriting over term rewriting, consider the rule $double(x) \to x + x$. In the term rewriting step



the subterm t is copied. If t is a large term, then this operation can be very costly in terms of space and time. Moreover, after this reduction step the value (normal form) of t has to be computed twice unless t is already in normal form. Therefore, this reduction step duplicates the work that is necessary to evaluate t. The usual way of avoiding these problems is to create two pointers to the subterm t instead of copying it.



In the resulting directed acyclic graph, the subterm t is *shared*. An evaluation of t in the directed acyclic graph corresponds to a parallel evaluation of the two occurrences of t in the term $t+t$. Thus representing expression as graphs instead of terms saves both time and space. Investigations of term graph rewriting include [19,4,18,10,11,1,14,15,2,17]. It has been shown by Kurihara and Ohuchi [11] that directed acyclic graphs can be represented by well-marked terms; thus graph transformations can be modeled by rewriting well-marked terms. We first recapitulate some basic notions. Most of them stem from [11].

Let $\mathcal{F}$ be a signature and $\mathcal{V}$ be a set of variables. Let M be a countably infinite set of objects called *marks* (we will use natural numbers as marks). Let $\mathcal{F}^* = \{f^\mu \mid f \in \mathcal{F}, \mu \in M\}$ be the set of *marked function symbols*. For all $f^\mu \in \mathcal{F}^*$, the arity of f^μ coincides with that of f. Moreover, we define $symbol(f^\mu) = f$ and $mark(f^\mu) = \mu$. Analogously, let $\mathcal{V}^* = \{x^\mu \mid x \in \mathcal{V}, \mu \in M\}$ be the set of *marked variables*, $symbol(x^\mu) = x$, and $mark(x^\mu) = \mu$. The set of *marked terms* over $\mathcal{F}^*$ and $\mathcal{V}^*$ is defined in the usual way and is denoted by $\mathcal{T}(\mathcal{F}^*, \mathcal{V}^*)$. The set of all marks appearing in a marked term $t^* \in \mathcal{T}(\mathcal{F}^*, \mathcal{V}^*)$ is denoted by $marks(t^*)$. The set $\mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$ of *well-marked terms* over $\mathcal{F}^*$ and $\mathcal{V}^*$ is the subset of $\mathcal{T}(\mathcal{F}^*, \mathcal{V}^*)$ such that $t^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$ if and only if, for every pair (t_1^*, t_2^*) of subterms of t^*, $mark(root(t_1^*)) = mark(root(t_2^*))$ implies $t_1^* = t_2^*$. For example, the term $0^1 +^0 0^1$ is well-marked but $0^1 +^1 0^1$ is not. Well-marked terms have an exact correspondence to labeled directed acyclic graphs; the reader is referred to [11] for details. In contrast to [11], we are solely interested in well-marked terms. Thus, throughout the whole paper, marked stands for well-marked. Two subterms t_1^* and t_2^* of a marked term t^* are shared in t^* if $t_1^* = t_2^*$; e.g. 0^1 and 0^1 are shared in $0^1 +^0 0^1$.

If t^* is a marked term, then $e(t^*)$ denotes the unmarked term obtained from t^* by erasing all marks. Two marked terms s^* and t^* are *bisimilar*[3] (denoted by $s^* \sim t^*$) if and only if $e(s^*) = e(t^*)$. The marks of a marked term s^* are called *fresh* w.r.t. another marked term t^* if $marks(s^*) \cap marks(t^*) = \emptyset$. A *marked substitution* $\sigma^* : \mathcal{V}^* \to \mathcal{T}(\mathcal{F}^*, \mathcal{V}^*)$ is a substitution which satisfies $x^\mu\sigma^* \sim y^\nu\sigma^*$ for all $x^\mu, y^\nu \in \mathcal{D}(\sigma^*)$ with $symbol(x^\mu) = symbol(y^\nu)$. This definition of marked substitution ensures that the unmarked substitution σ obtained from σ^* by erasing all marks is well-defined (i.e., σ really is a substitution). The notion *marked context*

[3] The origin of the notion "bisimilarity" is explained in [2].

is defined in the obvious way.

Definition 3.1 A rule $l^* \to r^* \Leftarrow c^*$ is a *marked version* of a rule $l \to r \Leftarrow c$ in $\mathcal{R}$ if $e(l^*) = l$, $e(r^*) = r$, $e(c^*) = c$, and, for all $x^\mu, y^\nu \in \mathcal{V}ar(l^* \to r^* \Leftarrow c^*)$, $symbol(x^\mu) = symbol(y^\nu)$ implies $mark(x^\mu) = mark(y^\nu)$.

The last condition can be rephrased as: every marked occurrence of a variable $x \in \mathcal{V}ar(l \to r \Leftarrow c)$ must have the same mark in $l^* \to r^* \Leftarrow c^*$. For the sake of simplicity, marks on variables in marked rewrite rules will be omitted from now on because these marks are unique anyway. So variables in rewrite rules are maximally shared. In contrast to [15], we do not use a minimal structure sharing scheme here (different structure sharing schemes are discussed in [11]). In that paper, in an application of a marked rule $l^* \to r^* \Leftarrow c^*$ all the marks on function symbols in r^*, c^*, and $x\sigma^*$ (for every extra variable x in $l \to r \Leftarrow c$) were assumed to be fresh and mutually distinct. Here $x\sigma^*$ gets fresh marks only if x is a free variable. If x is a bound variable (i.e., it occurs in some t_i^*), then $x\sigma^*$ is the marked term that is determined by the "match" $v_i^* \approx t_i^*\sigma^*$, where v_i^* is a reduct of $s_i^*\sigma$. Let us formally define such a "match".

Definition 3.2 Let v^* and t^* be marked terms with $\mathcal{V}ar(v^*) \cap \mathcal{V}ar(t^*) = \emptyset$. Suppose moreover that t^* is a linear constructor term. Let $\mathcal{V}ar(t^*) = \{x_1, \ldots, x_m\}$ and $p_1, \ldots, p_m \in \mathcal{P}os(t^*)$ such that $t^*|_{p_j} = x_j$. We say that t^* *matches* v^* via substitution σ^*, denoted by $v^* \approx t^*\sigma^*$, if $e(v^*) = e(t^*\sigma^*)$ and $x_j\sigma^* = v^*|_{p_j}$.

Now we are in a position to define the conditional term graph rewriting relation.

Definition 3.3 Let $\mathcal{R}$ be a right-stable oriented CTRS and s^*, t^* be marked terms. Let $\Rightarrow_{\mathcal{R}_0} = \emptyset$ and, for $n > 0$, define $s^* \Rightarrow_{\mathcal{R}_n} t^*$ if there exists a marked version $l^* \to r^* \Leftarrow s_1^* \to t_1^*, \ldots, s_k^* \to t_k^*$ of a conditional rewrite rule $\rho : l \to r \Leftarrow s_1 \to t_1, \ldots, s_k \to t_k$ from $\mathcal{R}$, a marked substitution σ^*, and a marked context $C^*[, \ldots ,]$ such that

(i) $s^* = C^*[l^*\sigma^*, \ldots, l^*\sigma^*]$ and $t^* = C^*[r^*\sigma^*, \ldots, r^*\sigma^*]$,

(ii) $l^*\sigma^*$ is not a subterm of $C^*[, \ldots ,]$,

(iii) for every $1 \le i \le k$, there are marked terms v_i^* such that
- $s_i^*\sigma^* \Rightarrow_{\mathcal{R}_{n-1}}^* v_i^* \approx t_i^*\sigma^*$ if t_i^* is a linear constructor term
- $s_i^*\sigma^* \Rightarrow_{\mathcal{R}_{n-1}}^* v_i^* \sim t_i^*$ if t_i^* is a ground $\mathcal{R}_u$-normal form

(iv) all marks on function symbols in r^*, s_i^*, t_i^*, and $x\sigma^*$ (for every free extra variable $x \in \mathcal{EV}ar^f(\rho)$) are mutually distinct and fresh w.r.t. s^*.

$l^*\sigma^*$ is called the *contracted marked redex* in s^*. We use the notation $s^* \Rightarrow_{\mathcal{R}_n}^{l^*\sigma^*} t^*$ in order to specify the contracted marked redex. Note that all shared subterms $l^*\sigma^*$ are replaced simultaneously by $r^*\sigma^*$.

If we are dealing with oriented CTRSs with normalizing conditions instead of oriented CTRSs, then it is further required in (iii) that $e(v_i^*)$ is a ground $\mathcal{R}_u$-normal form.

In order to illustrate how term graph rewriting works, let $\mathcal{R}$ be the CTRS

$$\begin{aligned} twoTimes(nil) &\to nil \\ twoTimes(x : xs) &\to x : x : twoTimes(xs) \\ fourTimes(nil) &\to nil \\ fourTimes(x : xs) &\to zs \Leftarrow twoTimes(x : xs) \to^! ys, twoTimes(ys) \to^! zs \end{aligned}$$

We have $fourTimes^0(0^2 :^1 nil^3) \Rightarrow_\mathcal{R} 0^2 :^{11} 0^2 :^{12} 0^2 :^{14} 0^2 :^{15} nil^{17}$ because

$$\begin{aligned} twoTimes^4(0^2 :^5 nil^3) &\Rightarrow_\mathcal{R} 0^2 :^6 0^2 :^7 twoTimes^8(nil^3) \\ &\Rightarrow_\mathcal{R} 0^2 :^6 0^2 :^7 nil^9 \end{aligned}$$

and

$$\begin{aligned} twoTimes^{10}(0^2 :^6 0^2 :^7 nil^9) &\Rightarrow_\mathcal{R} 0^2 :^{11} 0^2 :^{12} twoTimes^{13}(0^2 :^7 nil^9) \\ &\Rightarrow_\mathcal{R} 0^2 :^{11} 0^2 :^{12} 0^2 :^{14} 0^2 :^{15} twoTimes^{16}(nil^9) \\ &\Rightarrow_\mathcal{R} 0^2 :^{11} 0^2 :^{12} 0^2 :^{14} 0^2 :^{15} nil^{17} \end{aligned}$$

Note that in $0^2 :^{11} 0^2 :^{12} 0^2 :^{14} 0^2 :^{15} nil^{17}$ every element in the list is shared.

4 Adequacy

In this section it will be shown that conditional term graph rewriting is adequate for simulating conditional term rewriting in orthogonal right-stable 2-CTRSs and good-natured 3-CTRSs, respectively. More precisely, we will show that the mapping e which erases all marks from a well-marked term is an adequate mapping in the sense of Kennaway et al. [10], that is, it is surjective, preserves normal forms, preserves reductions, and is cofinal. Surjectivity ensures that every term can be represented as a directed acyclic graph (well-marked term). The normal form condition ensures that a graph is a final result of a computation if the term which it represents also is, and vice versa. Preservation of reduction ensures that every graph reduction sequence represents some term reduction sequence. Cofinality ensures that for every term rewriting computation, there is a term graph rewriting computation which can be mapped, not necessarily to the term rewriting computation, but to some extension of it.

Theorem 4.1 *Let $\mathcal{R}$ be an orthogonal right-stable oriented 2-CTRS. For all $n \in \mathbb{N}$, $\Rightarrow_{\mathcal{R}_n}$ is an adequate implementation of $\to_{\mathcal{R}_n}$, that is,*

(i) *e is surjective,*

(ii) *$\forall t^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$: $t^* \in NF(\Rightarrow_{\mathcal{R}_n})$ if and only if $e(t^*) \in NF(\to_{\mathcal{R}_n})$,*

(iii) *$\forall s^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$: if $s^* \Rightarrow^*_{\mathcal{R}_n} t^*$, then $e(s^*) \to^*_{\mathcal{R}_n} e(t^*)$,*

(iv) *$\forall s^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$: if $e(s^*) \to^*_{\mathcal{R}_n} u$, then there is a $t^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$ such that $s^* \Rightarrow^*_{\mathcal{R}_n} t^*$ and $u \to^*_{\mathcal{R}_n} e(t^*)$.*

Proof. We use induction on n.

(i) Surjectivity is obvious.

(ii) The *if* direction is easily shown. For an indirect proof of the *only if* direction, suppose $e(t^*) \notin NF(\to_{\mathcal{R}_n})$, i.e., $e(t^*) = C[l\sigma]\to_{\mathcal{R}_n}C[r\sigma]$ by using the rule $\rho : l \to r \Leftarrow s_1 \to t_1, \dots, s_k \to t_k$ at position p. So, for every $s_i \to t_i$, we have $s_i\sigma\to^*_{\mathcal{R}_{n-1}}t_i\sigma$. Let l^* and σ^* be marked version of l and σ such that $t^*|_p = l^*\sigma^*$ (note that l^* is left-linear). Let $l^* \to r^* \Leftarrow s_1^* \to t_1^*, \dots, s_k^* \to t_k^*$ be a marked version of ρ such that all marks on r^*, s_i^*, and t_i^* are fresh w.r.t. t^* and mutually distinct. Furthermore, σ^* is extended to $\mathcal{EV}ar(\rho)$ as follows: for all $z \in \mathcal{EV}ar^f(\rho)$ let $z\sigma^*$ be a marked version of $z\sigma$ such that all marks are mutually distinct and fresh w.r.t. t^*, r^*,s_i^*, and t_i^*. Let $C^*[,\dots,]$ be the marked context such that $t^* = C^*[l^*\sigma^*,\dots,l^*\sigma^*]$ and $l^*\sigma^*$ is not a subterm of $C^*[,\dots,]$. We extend σ^* successively to the bound variables $\mathcal{EV}ar^b(\rho)$. We have $e(s_1^*\sigma^*) = s_1\sigma\to^*_{\mathcal{R}_{n-1}} = t_1\sigma$. It follows from the inductive hypothesis that there exists a marked term v_1^* such that $s_1^*\sigma^* \Rightarrow^*_{\mathcal{R}_{n-1}} v_1^*$ and $t_1\sigma\to^*_{\mathcal{R}_{n-1}}e(v_1^*)$. If t_1 is a ground $\mathcal{R}_u$-normal form, then $t_1 = e(v_1^*)$ and hence $v_1^* \sim t_1^*$. Otherwise, if t_1 is a linear constructor term, it can be written as $t_1 = C_1[x_1,\dots,x_m]$, where all variables are displayed and pairwise distinct. It follows that $e(v_1^*) = C_1[w_1,\dots,w_m]$, where $x_i\sigma\to^*_{\mathcal{R}_{n-1}}w_i$. Hence v_1^* has the form $C_1^*[w_1^*,\dots,w_m^*]$. We define a substitution σ_1^* on $\mathcal{V}ar(l) \cup \mathcal{EV}ar^f(\rho) \cup \mathcal{V}ar(t_1)$ by $x\sigma_1^* = x\sigma^*$ for all $x \in \mathcal{V}ar(l) \cup \mathcal{EV}ar^f(\rho)$ and $x_i\sigma^* = w_i^*$ for $1 \le i \le m$. Clearly, $v_1^* \approx t_1^*\sigma_1^*$ and $s_2\sigma\to^*_{\mathcal{R}_{n-1}}e(s_2^*\sigma_1^*)$. Since $s_2\sigma\to^*_{\mathcal{R}_{n-1}}t_2\sigma$ as well, it follows from level-confluence of $\to_{\mathcal{R}}$ that there is a common reduct u_2 of $e(s_2^*\sigma_1^*)$ and $t_2\sigma$ w.r.t. $\to_{\mathcal{R}_{n-1}}$. Furthermore, there is a substitution τ_2 such that $u_2 = t_2\tau_2$ because t_2 is a linear constructor term or a ground $\mathcal{R}_u$-normal form (in the latter case τ is the empty substitution). Applying the inductive hypothesis to $e(s_2^*\sigma_1^*)\to^*_{\mathcal{R}_{n-1}}t_2\tau_2$ yields a marked term v_2^* such that $s_2^*\sigma_1^* \Rightarrow^*_{\mathcal{R}_{n-1}} v_2^*$ and $t_2\tau_2\to^*_{\mathcal{R}_{n-1}}e(v_2^*)$. As above, we can extend σ_1^* to $\mathcal{V}ar(t_2)$ yielding σ_2^*. By continuing along these lines, there exist v_i^* and an extension σ_k^* of σ_1^* to the bound variables $\mathcal{EV}ar^b(\rho) = \bigcup_{i=1}^k \mathcal{V}ar(t_k)$ such that $s_i^*\sigma_k^* \Rightarrow^*_{\mathcal{R}_{n-1}} v_i^* \approx t_i\sigma_k^*$. Therefore, $t^* \Rightarrow_{\mathcal{R}_n} C^*[r^*\sigma^*,\dots,r^*\sigma^*]$ which contradicts $t^* \in NF(\Rightarrow_{\mathcal{R}_n})$.

(iii) We proceed by induction on the length ℓ of $s^* \Rightarrow^*_{\mathcal{R}_n} t^*$. The base case $\ell = 0$ clearly holds. Thus consider $s^* \Rightarrow^{l^*\sigma^*}_{\mathcal{R}_n} u^* \Rightarrow^{\ell}_{\mathcal{R}_n} t^*$. According to the inductive hypothesis on ℓ, $e(u^*)\to^*_{\mathcal{R}_n}e(t^*)$. Since $s^* \Rightarrow^{l^*\sigma^*}_{\mathcal{R}_n} u^*$, we have $s^* = C^*[l^*\sigma^*,\dots,l^*\sigma^*]$, $l^*\sigma^*$ is not a subterm of $C^*[,\dots,]$, $u^* = C^*[r^*\sigma^*,\dots,r^*\sigma^*]$, and, for every $s_i^* \to t_i^*$, there is a marked term v_i^* such that $s_i^*\sigma^* \Rightarrow^*_{\mathcal{R}_{n-1}} v_i^* \approx t_i^*\sigma^*$. Let $\sigma = e(\sigma^*)$, i.e., $x\sigma = e(x\sigma^*)$ for all $x \in \mathcal{D}(\sigma^*)$. By the inductive hypothesis on n, $e(s_i^*)\sigma\to^*_{\mathcal{R}_{n-1}}e(v_i^*) = e(t_i^*)\sigma$. Hence $l\sigma\to_{\mathcal{R}_n}r\sigma$ and $e(s^*) \to^+_{\mathcal{R}_n} e(t^*)$.

(iv) We use induction on the length ℓ of $e(s^*)\to^{\ell}_{\mathcal{R}_n} u$. The proof is illustrated in Fig. 1. The case $\ell = 0$ is trivial. So we consider $e(s^*)\to^{\ell}_{\mathcal{R}_n} \bar{u}\to_{\mathcal{R}_n}u$. By the inductive hypothesis on ℓ, there exists a $\bar{t}^* \in \mathcal{T}_w(\mathcal{F}^*,\mathcal{V}^*)$ such that $s^* \Rightarrow^*_{\mathcal{R}_n} \bar{t}^*$

$$\begin{tikzcd}
e(s^*) \arrow[r, "\ell"', "{}_{\mathcal{R}_n}"] & \bar{u} \arrow[r, "{}_{\mathcal{R}_n}"'] \arrow[d, "*"', "\mathcal{R}_n"] & u \arrow[d, "*"', "\mathcal{R}_n"] & \\
 & e(\bar{t}^*) \arrow[r, "*"', "{}_{\mathcal{R}_n}"] & v \arrow[r, "*"', "{}_{\mathcal{R}_n}"] & e(t^*) \\
s^* \arrow[r, Rightarrow, "*"', "{}_{\mathcal{R}_n}"] & \bar{t}^* \arrow[rr, Rightarrow, "*"', "{}_{\mathcal{R}_n}"] & & t^*
\end{tikzcd}$$

Fig. 1. Proof of Theorem 4.1.

and $\bar{u} \rightarrow^*_{\mathcal{R}_n} e(\bar{t}^*)$. Let $\bar{t} = e(\bar{t}^*)$. Suppose $\bar{u} = C[l\sigma] \rightarrow_{\mathcal{R}_n} C[r\sigma] = u$ by using the rule $\rho : l \rightarrow r \Leftarrow s_1 \rightarrow t_1, \ldots, s_k \rightarrow t_k$ at the position p, i.e., $C[l\sigma]|_p = l\sigma$. By the parallel moves lemma for $\rightarrow_{\mathcal{R}_n}$, there is a $v \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ such that $u \mathrel{+\!\!\!\!\!\to}^*_{\mathcal{R}_n} v$ and $\bar{t} \mathrel{+\!\!\!\!\!\to}_{\mathcal{R}_n} v$. In particular, the redexes contracted in the step $\bar{t} \mathrel{+\!\!\!\!\!\to}_{\mathcal{R}_n} v$ are the descendants $p \backslash (\bar{u} \rightarrow^*_{\mathcal{R}_n} \bar{t})$ of p in $\bar{t}$. Let $Q = p \backslash (\bar{u} \rightarrow^*_{\mathcal{R}_n} \bar{t})$. Note that $Q \subseteq \mathcal{P}os(\bar{t})$ consists of pairwise independent positions. For every $q \in Q$, $\bar{t}^*|_q$ can be written as $\bar{t}^*|_q = l^*_q \tau^*_q$, where l^*_q is a marked version of l and τ^*_q is a marked substitution. As in the proof of (ii), one can show that $l^*_q \tau^*_q \Rightarrow_{\mathcal{R}_n} r^* \tau^*_q$. Let

$$Q' = \{q' \in \mathcal{P}os(\bar{t}^*) \mid \bar{t}^*|_{q'} = l^*_q \tau^*_q \text{ for some } q \in Q\}$$

Note that $Q \subseteq Q'$. It is not difficult to prove that Q' consists of pairwise independent positions. Let t^* be the marked term obtained from $\bar{t}^*$ by contracting all the redexes $l^*_q \tau^*_q$. Let $\tau_q = e(\tau^*_q)$. Since $\bar{t} \mathrel{+\!\!\!\!\!\to}_{\mathcal{R}_n} v$ by contracting the redexes in Q and $\bar{t} \mathrel{+\!\!\!\!\!\to}_{\mathcal{R}_n} e(t^*)$ by contracting the redexes in Q', it follows that $v \mathrel{+\!\!\!\!\!\to}_{\mathcal{R}_n} e(t^*)$ by contracting the redexes in $Q' \setminus Q$. All in all, $s^* \Rightarrow^*_{\mathcal{R}_n} t^*$ and $u \rightarrow^*_{\mathcal{R}_n} e(t^*)$.

□

Theorem 4.2 *If $\mathcal{R}$ is a good-natured 3-CTRS, then $\Rightarrow_{\mathcal{R}_n}$ is an adequate implementation of $\rightarrow_{\mathcal{R}_n}$ for every $n \in \mathbb{N}$.*

Proof. The proof is essentially the same as that of Theorem 4.1. Only the second paragraph of the proof of (ii) needs some modifications.

We extend σ^* successively to the bound variables $\mathcal{EV}ar^b(\rho)$. Since $e(s^*_1\sigma^*) = s_1\sigma \rightarrow^*_{\mathcal{R}_{n-1}} t_1\sigma$ and $t_1\sigma$ is a ground $\mathcal{R}_u$-normal form, it follows from the inductive hypothesis that there is a marked term v^*_1 such that $s^*_1\sigma^* \Rightarrow^*_{\mathcal{R}_{n-1}} v^*_1$ and $e(v^*_1) = t_1\sigma$. If t_1 is a ground $\mathcal{R}_u$-normal form, then $t_1\sigma = t_1$ and hence $v^*_1 \sim t^*_1$. Otherwise, if t_1 is a linear constructor terms, it can be written as $t_1 = C_1[x_1, \ldots, x_m]$, where all variables are displayed and pairwise distinct. It follows that $e(v^*_1) = C_1[w_1, \ldots, w_m]$, so that v^*_1 has the form $C^*_1[w^*_1, \ldots, w^*_m]$. We define σ^* on $\mathcal{V}ar(t_1)$ by $x_i\sigma^* = w^*_i$ for $1 \leq i \leq m$. Clearly, $v^*_1 \approx t^*_1\sigma^*$ and $e(s^*_2\sigma^*) = s_2\sigma$. By continuing along these lines, there exist v^*_i and an extension of σ^* to the bound variables $\mathcal{EV}ar^b(\rho)$ such that $s^*_i\sigma^* \Rightarrow^*_{\mathcal{R}_{n-1}} v^*_i \approx t^*_i\sigma^*$. Therefore, $t^* \Rightarrow_{\mathcal{R}_n} C^*[r^*\sigma^*, \ldots, r^*\sigma^*]$ which contradicts $t^* \in NF(\Rightarrow_{\mathcal{R}_n})$. □

There are the following corollaries for the classes of CTRSs under consideration.

Corollary 4.3 *$\Rightarrow_{\mathcal{R}}$ is an adequate implementation of $\rightarrow_{\mathcal{R}}$.*

It is a direct consequence of the preceding results that $\Rightarrow_{\mathcal{R}}$ is a sound and complete implementation of $\rightarrow_{\mathcal{R}}$ in the sense of Barendregt et al. [4]. Recall that soundness ensures that the graph implementation of a CTRS cannot give incorrect results, while completeness ensures that term graph rewriting gives all results.

Corollary 4.4 *$\Rightarrow_{\mathcal{R}}$ is a sound and complete implementation of $\rightarrow_{\mathcal{R}}$, i.e.,*

(i) $s^* \Rightarrow^*_{\mathcal{R}} t^* \in NF(\Rightarrow_{\mathcal{R}})$ *implies* $e(s^*) \rightarrow^*_{\mathcal{R}} e(t^*) \in NF(\rightarrow_{\mathcal{R}})$ *(soundness),*

(ii) $\forall s^* \in \mathcal{T}_w(\mathcal{F}^*, \mathcal{V}^*)$*: if* $e(s^*) \rightarrow^*_{\mathcal{R}} u \in NF(\rightarrow_{\mathcal{R}})$*, then there is a marked term* t^* *such that* $s^* \Rightarrow^*_{\mathcal{R}} t^* \in NF(\Rightarrow_{\mathcal{R}})$ *and* $e(t^*) = u$ *(completeness).*

Note that in the entire section, there is only one place at which we made use of the fact that $\mathcal{R}$ is orthogonal: Theorems 4.1 (iv) and 4.2 (iv) crucially depend on the fact that the parallel moves lemma holds for $\rightarrow_{\mathcal{R}_n}$. Since the parallel moves lemma remains valid if $\mathcal{R}$ is almost orthogonal, so do all of the preceding results if we replace orthogonality with almost orthogonality.

References

[1] Ariola, Z. and J. Klop, *Equational term graph rewriting*, Fundamenta Informaticae **26** (1996), pp. 207–240.

[2] Ariola, Z., J. Klop and D. Plump, *Bisimilarity in term graph rewriting*, Information and Computation **156** (2000), pp. 2–24.

[3] Avenhaus, J. and C. Loría-Sáenz, *On conditional rewrite systems with extra variables and deterministic logic programs*, in: *Proceedings of the 5th International Conference on Logic Programming and Automated Reasoning*, Lecture Notes in Artificial Intelligence **822** (1994), pp. 215–229.

[4] Barendregt, H., M. v. Eekelen, J. Glauert, J. Kennaway, M. Plasmeijer and M. Sleep, *Term graph rewriting*, in: *Proceedings of Parallel Architectures and Languages Europe*, Lecture Notes in Computer Science **259** (1987), pp. 141–158.

[5] Bergstra, J. and J. Klop, *Conditional rewrite rules: Confluence and termination*, Journal of Computer and System Sciences **32** (1986), pp. 323–362.

[6] Dershowitz, N. and M. Okada, *A rationale for conditional equational programming*, Theoretical Computer Science **75** (1990), pp. 111–138.

[7] Dershowitz, N., M. Okada and G. Sivakumar, *Canonical conditional rewrite systems*, in: *Proceedings of the 9th Conference on Automated Deduction*, Lecture Notes in Computer Science **310** (1988), pp. 538–549.

[8] Ganzinger, H. and U. Waldmann, *Termination proofs of well-moded logic programs via conditional rewrite systems*, in: *Proceedings of the 3rd International Workshop on Conditional Term Rewriting Systems*, Lecture Notes in Computer Science **656** (1993), pp. 113–127.

[9] Hanus, M., *On extra variables in (equational) logic programming*, in: *Proceedings of the 12th International Conference on Logic Programming* (1995), pp. 665–679.

[10] Kennaway, J., J. Klop, M. Sleep and F.-J. d. Vries, *On the adequacy of term graph rewriting for simulating term rewriting*, ACM Transactions on Programming Languages and Systems **16** (1994), pp. 493–523.

[11] Kurihara, M. and A. Ohuchi, *Modularity in noncopying term rewriting*, Theoretical Computer Science **152** (1995), pp. 139–169.

[12] Marchiori, M., *Unravelings and ultra-properties*, in: *Proceedings of the 5th International Conference on Algebraic and Logic Programming*, Lecture Notes in Computer Science **1139** (1996), pp. 107–121.

[13] Middeldorp, A. and E. Hamoen, *Completeness results for basic narrowing*, Applicable Algebra in Engineering, Communication and Computing **5** (1994), pp. 213–253.

[14] Ohlebusch, E., *A uniform framework for term and graph rewriting applied to combined systems*, Information Processing Letters **73** (2000), pp. 53–59.

[15] Ohlebusch, E., *Implementing conditional term rewriting by graph rewriting*, Theoretical Computer Science **262** (2001), pp. 311–331.

[16] Ohlebusch, E., "Advanced Topics in Term Rewriting," Springer-Verlag, 2002.

[17] Plump, D., *Term graph rewriting*, in: Ehrig, H., Engels, G., Kreowski, H.-J., and Rozenberg, G., editors, *Handbook of Graph Grammars and Computing by Graph Transformation*, vol. **2**, World Scientific (1999), pp. 3–61.

[18] Sleep, M., M. Plasmeijer and M. v. Eekelen, editors, "Term Graph Rewriting," John Wiley & Sons, 1993.

[19] Staples, J., *Computation on graph-like expressions*, Theoretical Computer Science **10** (1980), pp. 171–185.

[20] Suzuki, T., A. Middeldorp and T. Ida, *Level-confluence of conditional rewrite systems with extra variables in right-hand sides*, in: *Proceedings of the 6th International Conference on Rewriting Techniques and Applications*, Lecture Notes in Computer Science **914** (1995), pp. 179–193.