Università di Pisa

Dipartimento di Informatica

Technical Report : TR-14-97

(Cyclic) Term Graph Rewriting is adequate for Rational Parallel Term Rewriting

Andrea Corradini, Frank Drewes

July 16, 1997

ADDR: Corso Italia 40,56125 Pisa,Italy. TEL: +39-50-887111. FAX: +39-50-887226

(Cyclic) Term Graph Rewriting is adequate for Rational Parallel Term Rewriting*

Andrea Corradini[1] and Frank Drewes[2]

[1] Università di Pisa, Dipartimento di Informatica, Corso Italia 40, 56125 Pisa, Italy (`andrea@di.unipi.it`)
[2] Universität Bremen, Fachbereich Mathematik und Informatik, Postfach 330440, 28334 Bremen, Germany (`drewes@informatik.uni-bremen.de`)

Abstract. *Acyclic* Term Graphs are able to represent terms with sharing, and the relationship between Term Graph Rewriting (TGR) and Term Rewrtiting (TR) is now well understood [BvEG+87, HP91]. During the last years, some researchers considered the extension of TGR to possibly *cyclic* term graphs, which can represent possibly infinite, rational terms. In [KKSdV94] the authors formalize the classical relationship between TGR and TR as an "adequate mapping" between rewriting systems, and extend it by proving that unraveling is an adequate mapping from cyclic TGR to rational, infinitary term rewriting: In fact, a single graph reduction may correspond to an infinite sequence of term reductions.
Using the same notions, we propose a different adequacy result, showing that unraveling is an adequate mapping from cyclic TGR to *rational parallel term rewriting*, where at each reduction infinitely many rules can be applied in parallel. We also argue that our adequacy result is more natural than that proposed in [KKSdV94], because the length of the reduction sequences is preserved by unraveling, and collapsing rules are treated in a completely uniform way.

1 Introduction

The classical theory of *Term Graph Rewriting* studies the issue of representing finite terms with directed, acyclic graphs, and of modeling term rewriting via graph rewriting (among the many contributions to this theory, we cite [Sta80, BvEG+87, Ken91, HP91, CR93]). The main advantage of using graphs is that the sharing of common subterms can be represented explicitly in a graph. Therefore the rewriting process is speeded up, because the rewriting steps do not have to be repeated for each copy of an identical subterm. For example, the rewrite rule $R : f(x) \to g(x)$ can be applied twice to term $t = k(f(a), r(f(a)))$, yielding in two steps term $t' = k(g(a), r(g(a)))$. If instead t is represented as a graph, and the two identical subterms are shared (as in graph G of Figure 1), then a single

* Research partly supported by the EC TMR Network GETGRATS (General Theory of Graph Transformation Systems) through the Dipartimento di Informatica, Pisa, and the University of Bremen.

application of the rule is sufficient to reduce it to graph G' of Figure 1, which clearly represents term t'. Thus a single graph reduction may correspond to n term reductions, where n is the "degree of sharing" of the reduced subterm.

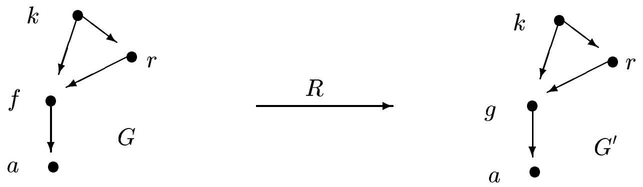

Fig. 1. An example of term graph rewriting

The various approaches to TGR present equivalent results for what concerns the speed-up of term rewriting due to the explicit sharing, but differ for the the graph rewriting formalism used. In particular, the approach that we call *operational*, introduced in [BvEG+87], defines a term graph reduction with a three-phase construction, which only makes sense for graphs that are intended to represent terms. On the contrary, the *algebraic approach* used in [HP91, CR93] uses the so-called "double-pushout approach" ([Ehr87]), which is a general graph rewriting formalism, and applies it to a category of term graphs.[3]

During the last years, some authors considered the extension of term graph rewriting to the cyclic case, allowing (finite, directed) cyclic graphs as well ([FW91, FRW90, KKSdV94]). The first consequence of this extension is that infinite terms (or, more precisely, *rational* terms, i.e., infinite terms with a finite number of distinct subterms) can be represented as well, exploiting cycles. The second effect is that a single graph reduction may now correspond to some infinite term rewriting. Consider for example rule R above: by applying it to graph H of Figure 2 one obtains the graph H'. Clearly, H represents the infinite term $f^\omega = f(f(f(\ldots)))$, while H' represents term g^ω. There are (at least) two possible ways of interpreting the rewriting of term f^ω to term g^ω via some number of applications of rule R:

1. g^ω is the limit of an infinite sequence of applications of R, i.e., $f^\omega \to_R g(f^\omega) \to_R g(g(f^\omega)) \to_R \ldots \rightsquigarrow_\omega g^\omega$.
2. g^ω is the result of the simultaneous application of R to an infinite number of redexes in f^ω: in a single step all the occurrences of f in f^ω are replaced by g.

[3] A variation of the algebraic approach making use of a single pushout construction is used for example in [Ken91]. Other differences among the various proposal for TGR concern the way terms are represented as graphs, but these are a mere syntactical issue (see [CMR+91]).

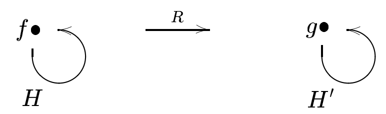

Fig. 2. An example of cyclic term graph rewriting

The interesting fact is that, unlike the above example, there are cases where these two interpretations lead to different results. This happens when *collapsing* rules are considered, i.e., rules having a variable as right-hand side. The most famous collapsing rule is the rule for identity, $R_I : I(x) \to x$, and the pathological case (considered already by many authors) is the application of R_I to I^ω. Using the first interpretation above, we have that $I^\omega \to_{R_I} I^\omega \to_{R_I} \ldots$, and clearly the limit of this sequence is I^ω itself. On the other hand, if we follow the second interpretation, all the occurrences of I in I^ω are deleted in a single step, and thus we should obtain as result a term containing no function symbols at all, i.e., some sort of "undefined" term.

It is worth stressing that *both* interpretations are meaningful from the point of view of cyclic term graph rewriting, and they correspond to two different choices of the graph rewriting algorithm. In fact, if one uses the operational approach to TGR, the "circular-I" (i.e., the graph having a single node labelled by I and one loop, which clearly unravels to I^ω) reduces via R_I to itself: and this is consistent with the first interpretation. This is the approach followed in [FW91, FRW90, DKP89, DK89, KKSdV94], where the authors elaborated a theory of (rational) *transfinite term rewriting*, showing that finite, cyclic graph rewriting is an adequate implementation for it. In essence, a finite graph derivation sequence has the "same effect" of a strongly convergent transfinite term rewriting sequence. As far as the notion of convergence is concerned, the well-known topological structure of (possibly infinite) terms is used, which, equipped with a suitable notion of distance, form a complete ultra-metric space [AN80]. However, it is worth stressing that this adequacy result treats circular redexes in an *ad hoc* way, because the term reduction sequence corresponding to the reduction of a circular redex is not strongly convergent (shortly, the depth of the reduced redexes does not tend to infinity). The authors propose essentially two solutions: either to stick to "almost noncollapsing" systems, having at most a single collapsing rule, or to impose an equivalence relation on terms which identifies all "hypercollapsing towers", i.e., infinite subterms made only of redexes of collapsing rules.

If instead one uses the algebraic approach to TGR, then the circular-I rewrites via R_I to a graph consisting of a single node. Although this fact has been stressed by other authors, as far as we know the present paper is the first attempt to provide a formal explanation of this phenomenon using (a suitable extension of) the theory of term rewriting. From an intuitive point of view, the second of the above interpretations must be used, assuming that all the occurrences of the operator I in I^ω are deleted in a single step by an infinite number of simultaneous applications of rule R_I. In this case the theory of transfinite term rewriting is no more helpful, because it cannot justify this result. We need instead the notion of "infinite parallel rewriting", that has been introduced by the first author in [Cor93]. Such an extension of term rewriting has been defined by exploiting the complete partial ordered structure of CT_Σ, the collection of possibly infinite, possibly partial terms over signature Σ. In particular, the result of the application of R_I in parallel to the infinitely many redexes in term I^ω is the undefined term $\bot$, which is the bottom element of CT_Σ.

As the main result of this paper we will show that for orthogonal systems, (cyclic) term graph rewriting (using the algebraic approach) is adequate for rational parallel term rewriting, using the notion of adequacy proposed in [KKSdV94]. There are two aspects that make, in our view, our result more appealing than the corresponding results in the mentioned paper. The first one is that the reduction of a cyclic term graph corresponds to a *single* parallel reduction of the corresponding term (possibly involving an infinite set of disjoint redexes): Thus the adequate mapping preserves the length of reduction sequences. Secondly, collapsing rules and circular redexes are treated in a completely uniform way thanks to the different algebraic structure exploited on terms. In particular, all hypercollapsing towers, which are artificially made equivalent in the other approach, can be reduced to the same term, $\bot$, by a single parallel reduction.

The paper is organized as follows. In Section 2 we summarize the basic definitions about infinite terms [GTWW77], orthogonal term rewriting [HL91], and parallel term rewriting [Cor93]. In Section 3 we introduce (possibly cyclic) term graphs, and we make precise their relationship with (sets of) rational terms, via the unraveling function. Algebraic term graph rewriting is the topic of Section 4, where we recall the basics of the double-pushout approach [Ehr87], apply it to the category of term graphs, and provide an encoding of term rewrite rules as graph rules. The main results of the paper are in Section 5. We first prove that a single reduction of a graph induces on the unraveled term a possibly infinite parallel reduction. Next this fact is used as a major lemma in the proof that the unraveling function is an adequate mapping from any orthogonal TGRS system to the orthogonal TRS obtained by unraveling its rules, provided that only rational terms and rational parallel reduction sequences are considered. Finally in Section 6 we summarize our contribution and hint new research topics.

2 Infinite terms and parallel term rewriting

In this background section we first introduce the algebra CT_Σ of possibly partial, possibly infinite terms. The collection of such terms forms a complete partial ordering which has been studied in depth in [GTWW77]. Next we introduce the basic definitions related to (orthogonal) term rewriting, which apply to infinite terms as well. Finally, we introduce the definition of infinite parallel rewriting, which exploits the CPO structure of terms according to [Cor93].

2.1 Infinite Terms

Most of the following definitions are borrowed from [GTWW77].

Let ω^* be the set of all finite strings of positive natural numbers. Elements of ω^* are called **occurrences**. The empty string is denoted by λ, and $u \leq w$ indicates that u is a prefix of w. Occurrences u, w are called **disjoint** (written $u|w$) if neither $u \leq w$ nor $w \leq u$.

Let Σ be a (one-sorted) signature, i.e., a ranked alphabet of operator symbols $\Sigma = \cup_{n \in \mathbb{N}} \Sigma_n$ and let X be a set of variables. A **term** over (Σ, X) is a partial function $t : \omega^* \to \Sigma \cup X$, such that the domain of definition of t, $\mathcal{O}(t)$, satisfies the following (where $w \in \omega^*$ and all $i \in \omega$):

- $wi \in \mathcal{O}(t) \Rightarrow w \in \mathcal{O}(t)$
- $wi \in \mathcal{O}(t) \Rightarrow t(w) \in \Sigma_n$ for some $n \geq i$.

$\mathcal{O}(t)$ is called the **set of occurrences** of t.

Given an occurrence $w \in \omega^*$ and a term t, the **subterm** of t at (occurrence) w is the term t/w defined as $t/w(u) = t(wu)$ for all $u \in \omega^*$. A term t is **finite** if $\mathcal{O}(t)$ is finite; it is **total** if $t(w) \in \Sigma_n \Rightarrow wi \in \mathcal{O}(t)$ for all $0 < i \leq n$; it is **linear** if no variable occurs more than once in it; and it is **rational** if it has a finite number of different subterms. Given terms t, s and an occurrence $w \in \omega^*$, the **replacement** of s in t at (occurrence) w, denoted $t[w \leftarrow s]$, is the term defined as $t[w \leftarrow s](u) = t(u)$ if $w \not\leq u$ or $t/w = \bot$, and $t[w \leftarrow s](wu) = s(u)$ otherwise.

The set of terms over (Σ, X) is denoted by $CT_\Sigma(X)$ (CT_Σ stays for $CT_\Sigma(\emptyset)$). Throughout the paper we will often use (for finite terms) the equivalent and more usual representation of terms as operators applied to other terms. Partial terms are made total in this representation by introducing the undefined term $\bot$ (called **bottom**), which represents the empty function $\bot : \emptyset \to \Sigma \cup X$. Thus, for example, if $x \in X$, $t = f(\bot, g(x))$ is the term such that $\mathcal{O}(t) = \{\lambda, 2, 2 \cdot 1\}$, $t(\lambda) = f \in \Sigma_2$, $t(2) = g \in \Sigma_1$, and $t(2 \cdot 1) = x \in X$.

It is well known that $CT_\Sigma(X)$ forms a **complete partial order** with respect to the "approximation" relation. We say that t **approximates** t' (written $t \leq t'$) iff t is less defined than t' as partial function. The least element of $CT_\Sigma(X)$ with respect to $\leq$ is clearly $\bot$. An ω**-chain** $\{t_i\}_{i<\omega}$ is an infinite sequence of terms $t_0 \leq t_1 \leq \ldots$. Every ω-chain $\{t_i\}_{i<\omega}$ in $CT_\Sigma(X)$ has a **least upper bound** (lub) $\bigcup_{i<\omega}\{t_i\}$ characterized as follows:

$$t = \bigcup_{i<\omega}\{t_i\} \quad \Leftrightarrow \quad \forall w \in \omega^* . \exists i < \omega . \forall j \geq i . t_j(w) = t(w)$$

Moreover, every pair of terms has a greatest lower bound. All this amounts to say that $CT_\Sigma(X)$ is an ω**-complete lower semilattice**, or simply a CPO.

2.2 Term Rewriting

We recall here the basic definitions of (orthogonal) term rewriting [HL91], which apply to infinite terms as well.

Let X and Y be two sets of variables. A **substitution** (from X to Y) is a function $\sigma : X \to CT_\Sigma(Y)$ (used in postfix notation). Such a substitution σ can be extended in a unique way to a continuous (i.e., monotonic and lub-preserving) function $\sigma : CT_\Sigma(X) \to CT_\Sigma(Y)$, which extends σ as follows

- $\perp\sigma = \perp$,
- $f(t_1, ..., t_n)\sigma = f(t_1\sigma, ..., t_n\sigma)$,
- $\left(\bigcup_{i<\omega}\{t_i\}\right)\sigma = \bigcup_{i<\omega}\{t_i\sigma\}$.

A **rewrite rule** $R = (l, r)$ is a pair of total terms of $CT_\Sigma(X)$, where $var(r) \subseteq var(l)$, l is finite and it is not a variable.[4] Terms l and r are called the **left-** and the **right-hand side** of R, respectively. A rule is called **left-linear** if l is linear, and it is **collapsing** if r is a variable. A **term rewriting system** (shortly **TRS**) $\mathcal{R}$ is a finite set of rewrite rules, $\mathcal{R} = \{R_i\}_{i\in I}$.

Given a term rewriting system $\mathcal{R}$, a **redex** (for REDucible EXpression) Δ of a term t is a pair $\Delta = (w, R)$ where $R : l \to r \in \mathcal{R}$ is a rule, and w is an occurrence of t, such that there exists a substitution σ which **realizes** Δ, i.e., such that $t/w = l\sigma$. In this case we say that t **reduces** (via Δ) to the term $t' = t[w \leftarrow r\sigma]$, and we write $t \to_\Delta t'$ or simply $t \to t'$. A **reduction sequence** $t_1 \to t_2 \to \ldots t_n$ is a finite sequence of reductions.

A TRS $\mathcal{R}$ is **orthogonal** (shortly, it is an **OTRS**) if all the rules in $\mathcal{R}$ are left-linear and it is **non-overlapping**; that is, the left-hand side of each rule does not unify with a non-variable subterm of any other rule in $\mathcal{R}$, or with a proper, non-variable subterm of itself. In this paper we will be concerned with orthogonal TRS's only.

2.3 Parallel Term Rewriting

As hinted in the introduction, (infinite) parallel rewriting is in our opinion the natural counterpart in the world of terms of the reduction of a cyclic graph. The definitions below summarize, in a simplified way, those in [Cor93]. Intuitively, *finite* parallel rewriting can be defined easily by exploiting the confluence of orthogonal term rewriting. In fact, the parallel reduction of a finite number of redexes is defined simply as any *complete development* of them: Any such development ends with the same term, so the result is well defined. Let us recall the relevant definitions.

Given two redexes of a term, the reduction of one of them can transform the other in various ways. The second redex can be destroyed, it can be left intact, or it can be copied a number of times. This is captured by the definition of residuals. We assume here that the rules belong to an OTRS, thus two redexes in a term are either the same or do not overlap.

[4] The restriction to left-finite rules can be motivated intuitively, as in [KKSdV94], by the requirement of checking in finite time the applicability of a rule to a term. On a more technical ground, in [Cor93] it is shown that point 2 of Theorem 6 below does not hold for left-infinite rules.

Definition 1 (residuals). Let $\Delta = (w, R)$ and $\Delta' = (w', R' : l' \to r')$ be two redexes in a term t. The **set of residual of** Δ **by** Δ' is denoted by $\Delta\backslash\Delta'$, and it is defined as

$$\Delta\backslash\Delta' = \begin{cases} \emptyset & \text{if } \Delta = \Delta' \\ \{\Delta\} & \text{if } w \not\geq w' \\ \{(w'w_x u, R) \mid r'/w_x = l'/v_x\} & \text{if } w = w'v_x u \text{ and } l'/v_x \text{ is a variable} \end{cases}$$

If Φ is a finite set of redexes of t and Δ is a redex of t, then the set of residuals of Φ by Δ, denoted $\Phi\backslash\Delta$, is defined as the union of $\Delta'\backslash\Delta$ for all $\Delta' \in \Phi$.

If Φ is a set of redexes of t and $s = (t \to_{\Delta_1} t_1 \ldots \to_{\Delta_n} t_n)$ is a reduction sequence, then $\Phi\backslash s$ is defined as Φ if $n = 0$, and as $(\Phi\backslash\Delta_1)\backslash s'$, where $s' = (t_1 \to_{\Delta_2} t_2 \ldots \to_{\Delta_n} t_n)$, otherwise. □

In the last definition, if $t \to_\Delta t'$ and Δ' is a redex of t, the orthogonality of the system ensures that every member of $\Delta'\backslash\Delta$ is a redex of t'.

Definition 2 (complete development). Let Φ be a finite set of redexes of t. A **development of** Φ is a reduction sequence such that after each initial segment s, the next reduced redex is an element of $\Phi\backslash s$. A **complete development of** Φ is a development s such that $\Phi\backslash s = \emptyset$. □

The following well-known fact is an easy consequence of the *parallel moves lemma* (see for example Lemma 2.2 in [HL91]).

Proposition 3. *All complete developments s and s' of a finite set of redexes Φ in a term t are finite, and end with the same term. Moreover, for each redex Δ of t, it holds $\Delta\backslash s = \Delta\backslash s'$. Therefore we can safely denote by $\Delta\backslash\Phi$ the residuals of Δ by any complete development of Φ (and similarly replacing Δ with a set of redexes Φ' of t).* □

Exploiting this fact, we define the parallel reduction of a finite set of redexes as any complete development of them.

Definition 4 (finite parallel redex reduction). A **parallel redex** Φ of a term t is a (possibly infinite, necessarily countable) set of distinct redexes in t. Given a *finite* parallel redex Φ of t, we write $t \to_\Phi t'$ and say that there is a **(finite) parallel reduction** from t to t' if there exists a complete development $t \to_{\Delta_1} t_1 \ldots \to_{\Delta_n} t'$ of Φ. □

We are now ready to extend the definition of application of parallel redexes to the infinite case. In [KKSdV94] it is shown that the notion of complete development can be defined for an infinite set of redexes as well, but that in order to ensure uniqueness of the resulting term, the corresponding possibly infinite reduction sequence must be "strongly convergent": This implies that not every infinite set of redexes has a complete development.

Thus we take a different approach. Given an infinite parallel redex Φ (i.e., an infinite set of redexes) of a term t, we consider a chain of approximations of

t, $t_0 \leq t_1 \leq t_2 \ldots$, such that their limit is t, and that only a finite subset of Φ applies to each t_i. For each $i < \omega$, let Φ_i be the finite subset of Φ containing all and only those redexes of t which are also redexes of t_i, and call d_i the result of the parallel reduction of Φ_i, i.e., $t_i \to_{\Phi_i} d_i$. Then the crucial fact is that the sequence of terms $d_0, d_1, d_2, \ldots$ defined in this way forms a chain: by definition we say that there is an infinite parallel reduction from t to $d = \bigcup_{i<\omega} d_i$ via Φ, written $t \to_{\Phi} d$. Here is the formal definition.

Definition 5 (parallel redex reduction). Given an infinite parallel redex Φ of a term t, let $t_0 \leq t_1 \leq \ldots t_n \leq \ldots$ be any chain approximating t (i.e., such that $\bigcup_{i<\omega}\{t_i\} = t$) and such that:

- for each $i < \omega$, every redex $(w, R) \in \Phi$ is either a redex of t_i or $t_i(w) = \perp$. That is, the image of the lhs of every redex in Φ is either all in t_i, or it is outside, but does not "cross the boundary".
- for each $i < \omega$, let $\Phi_i \subseteq \Phi$ be the subset of all redexes in Φ which are also redexes of t_i. Then Φ_i must be finite for all $i < \omega$.

For each $i < \omega$, let d_i be the result of the (finite) parallel reduction of t_i via Φ_i (i.e., $t_i \to_{\Phi_i} d_i$). Then we say that there is an **(infinite) parallel reduction** from t to $d \stackrel{def}{=} \bigcup_{i<\omega}\{d_i\}$ via Φ, and we write $t \to_{\Phi} d$. $\square$

The main result of this section states that the last definition is well-given. For a proof we refer to [Cor93] (Theorem 32).

Theorem 6 (parallel redex reduction is well-defined). *In the hypotheses of Definition 5:*

1. $\forall\, 0 \leq i < j < \omega \,.\, d_i \leq d_j$, *thus* $\{d_i\}_{i<\omega}$ *is a chain.*
2. *Definition 5 is well-given i.e., the result of the infinite parallel reduction of t via Φ does not depend on the choice of the chain approximating t, provided that it satisfies the required conditions.* $\square$

As an example, let us consider again the two infinite reductions mentioned in the introduction, according to the second of the two interpretations. In the term reduction from f^ω to g^ω (corresponding to the graph reduction of Figure 2), there are infinitely many redexes of rule $R : f(x) \to g(x)$ in the term f^ω, namely at occurrences $\lambda, 1, 1 \cdot 1, \ldots$. Let $\Phi = \{\Delta_n \stackrel{def}{=} (1^n, R) \mid n \in \mathbb{N}\}$ be this infinite parallel redex of f^ω. As for the chain of terms approximating $t = f^\omega$, let us choose $t_0 = \perp, t_1 = f(\perp), \ldots, t_n = f^n(\perp)$. Clearly, for each n the set $\Phi_n \subseteq \Phi$ of redexes of t_n contains exactly n redexes. For each n we have $t_n = f^n(\perp) \to_{\Phi_n} g^n(\perp)$, and thus, according to the definition, the result of the parallel reduction of f^ω via Φ is $\bigcup_{i<\omega}\{g^i(\perp)\} = g^\omega$.

In the case of rule R_I and of the circular I, choosing a similar approximating chain we have $\Phi = \{\Delta_n \stackrel{def}{=} (1^n, R_I) \mid n \in \mathbb{N}\}$, $t_n = I^n(\perp)$ for each n, $t_n = I^n(\perp) \to_{\Phi_n} \perp$, and thus I^ω reduces by Φ to $\bigcup_{i<\omega}\{\perp\} = \perp$.

We shall need the following easy result.

Proposition 7 (strong confluence of parallel reduction). *Given an orthogonal TRS $\mathcal{R}$, parallel reduction is strong confluent, i.e., if $t' \,{}_{\Phi'}\!\!\leftarrow t \rightarrow_{\Phi} t''$, then there exist t''', Ψ, Ψ' such that $t' \rightarrow_{\Psi'} t''' \,{}_{\Psi}\!\!\leftarrow t''$. As a consequence, parallel reduction is confluent.* □

3 Term Graphs and Rational Terms

We summarize here the definition of *term graphs* (or simply *graphs*), and their relationship with rational terms, as introduced in [KKSdV94]. However, since we will apply to those graphs the algebraic approach to graph rewriting, we shall slightly adapt the definition to our framework, emphasizing the categorical structure of the collection of graphs.

Term graphs are obtained from the usual representation of terms with sharing as *dag's* (*directed acyclic graph*), by dropping the acyclicity requirement. In such a way, a finite cyclic graph may represent a possibly infinite, but rational term. An equivalent definition could be given by using the so-called *jungles* used for example in [HP91, CR93], and dropping the restriction to acyclic structures. Since the two options are equivalent (in a very precise sense: the corresponding categories are equivalent, see [CMR$^+$91]), we preferred the more traditional presentation, in order to ease the comparison of our results with those in [KKSdV94].

Definition 8 (term graphs). Let Σ be a fixed, one-sorted[5] signature. A **(term) graph** G (over Σ) is a triple $G = (N_G, s_G, l_G)$, where

- N_G is a *finite* set of **nodes**,
- $s_G : N_G \rightarrow N_G^*$ is a partial function, called the **successor** function,
- $l_G : N_G \rightarrow \Sigma$ is a partial function, called the **labelling** function.

Moreover, it is required that s_G and l_G are defined on the same subset of N_G, and that for each node $n \in N_G$, if $l_G(n)$ is defined and it is an operator of arity k, then $s_G(n)$ has length exactly k. □

Definition 9 (morphisms, category of term graphs). A **(graph) morphism** $f : G \rightarrow H$ between two graphs G and H is a function $f : N_G \rightarrow N_H$, which preserves labelling and successor functions, i.e., for each $n \in N_G$, if $l_G(n)$ is defined, then $l_H(f(n)) = l_G(n)$ and $s_H(f(n)) = f^*(s_G(n))$ (where f^* is the obvious extension of f to lists of nodes).

The composition of graph morphisms is defined in the obvious way, and it is clearly associative; moreover, the identity function on nodes is a morphism, and therefore term graphs (over Σ) and their morphisms form a category that will be denoted by $\mathbf{TGraph}_\Sigma$. □

Thanks to the conditions imposed on term graphs in Definition 8, a term of CT_Σ can be *extracted* or *unraveled* from every node of a graph.

[5] The generalization to many-sorted signatures is straightforward, by labeling nodes with pairs ⟨*operator, sort*⟩, with the obvious meaning.

Definition 10 (from term graphs to terms and backwards). A **path** π in a graph $G = (N_G, s_G, l_G)$ from node n to node n' is a finite sequence $\pi = \langle n_1, j_1, n_2, j_2, \ldots, j_k, n_{k+1}\rangle$, where all j_i are natural numbers, all n_i are nodes, and such that $n_1 = n$, $n_{k+1} = n'$, and for all $1 \le i \le k$, $s_G(n_i)|_{j_i} = n_{i+1}$ (here $s|_i$ denotes the i-th element of the sequence s). It follows that there exists exactly one **empty path** '$\langle n\rangle$' from each node n to itself. The **occurrence of a path** $\pi = \langle n_1, j_1, n_2, j_2, \ldots, j_k, n_{k+1}\rangle$ is the list of natural numbers $j_1 \cdot j_2 \cdots j_k$, and it is denoted by $\mathcal{O}(\pi)$. Thus $\mathcal{O}(\pi) = \lambda$ iff π is an empty path. Clearly, for each node n there is at most one path in G having a given occurrence w.

A graph G is **acyclic** if there are no non-empty paths from one node to itself; graph G is a **tree** with root $\underline{n} \in N_G$ iff there exists exactly one path from $\underline{n}$ to any other node of G.

Let $G = (N_G, s_G, l_G)$ be a term graph. The set $var(G) \subseteq N_G$ of **variable nodes** or **empty nodes** of G is the set of nodes on which the labeling function (and thus also the successor function) is undefined. For each node $n \in N_G, \mathcal{U}_G[n]$, the **unraveling of** G **at** n is the term defined as

$$\mathcal{U}_G[n](w) = \begin{cases} n' & \text{if there is a path } \pi \text{ from } n \text{ to } n' \text{ with } \mathcal{O}(\pi) = w \\ & \text{and } n' \in var(G) \\ l_G(n') & \text{if there is a path } \pi \text{ from } n \text{ to } n' \text{ with } \mathcal{O}(\pi) = w \\ & \text{and } n' \notin var(G) \\ \perp & \text{otherwise.} \end{cases}$$

for all $w \in \omega^*$. It follows immediately from these definitions that for each $n \in N_G$ and for each occurrence w, $\mathcal{U}_G[n]/w = \mathcal{U}_G[n']$ iff there exists a path π from n to n' with $\mathcal{O}(\pi) = w$. By $\mathcal{U}[G]$ we denote the set $\mathcal{U}[G] = \{\mathcal{U}_G[n] \mid n \in N_G\}$.

Conversely, let T be a finite set of rational terms, and let us denote by $\overline{T}$ its closure under the subterm relation (i.e., $\overline{T} = \{t \mid t \text{ is a subterm of some term in } T\}$). Then the **term graph representation of** T, denoted $\mathcal{G}[T]$, is the graph $\mathcal{G}[T] = (N_{\mathcal{G}[T]}, s_{\mathcal{G}[T]}, l_{\mathcal{G}[T]})$ defined as follows:

1. $N_{\mathcal{G}[T]} = \overline{T}$;
2. $s_{\mathcal{G}[T]}(t) = \langle t_1, \ldots, t_k\rangle$ if $t = f(t_1, \ldots, t_k)$, and undefined if t is a variable;
3. $l_{\mathcal{G}[T]}(t) = f$ if $t = f(t_1, \ldots, t_k)$, and undefined if t is a variable. □

It is quite easily seen that, for every term graph G and node $n \in N_G$, $\mathcal{U}_G[n]$ is a well-defined and total term containing variables in $var(G)$ (thus $\mathcal{U}_G[n] \in CT_\Sigma(var(G))$). Such term can be infinite (because in a cyclic graph there can be infinitely many paths starting from one node), but since G is finite by definition, $\mathcal{U}_G[n]$ is necessarily a rational term, because the number of distinct subterms is bounded by the cardinality of N_G. It is worth noting also that if G is a tree with root $\underline{n}$, then $\mathcal{U}_G[\underline{n}]$ is a finite and linear term.

For what concerns the definition of $\mathcal{G}$, it is easy to verify that $\mathcal{G}[T]$ is a well-defined term graph ($\overline{T}$ is finite by the assumptions on T), such that $var(\mathcal{G}[T])$ consists of all the variables in T, and such that $\mathcal{U}[\mathcal{G}[T]] = \overline{T}$. There are in general many non-isomorphic graphs satisfying this last properties (i.e., graphs G such that $\mathcal{U}[G] = \overline{T}$): The representation proposed above is the most compact one, and it is called *fully-collapsed* in [HP91]. It is worth stressing that the results in Section 5 are independent of the chosen representation.

The above definitions made clear the relationship between objects of category $\mathbf{TGraph}_\Sigma$ and terms in CT_Σ. Such relationship can be extended to arrows of $\mathbf{TGraph}_\Sigma$ and to term substitutions as follows.

Proposition 11 (from morphism to substitutions and backwards). *Let $f : G \to H$ be a term graph morphism. The* **substitution induced by** *f is the substitution $\sigma_f : var(G) \to CT_\Sigma(var(H))$, defined as $x\sigma_f = \mathcal{U}_H[f(x)]$ for all $x \in var(G)$. Moreover the following hold:*

1. *If $f : G \to H$ is a morphism, then $\sigma_f \circ \mathcal{U}_G = \mathcal{U}_H \circ f$.*
2. *Let G be a tree with root $\underline{n}$ and let $\mathcal{U}_G[\underline{n}] = t$. Then for every graph H and every node $n' \in N_H$ with $t' = \mathcal{U}_H[n']$, there is a substitution σ such that $t\sigma = t'$* if and only if *there is a morphism $f : G \to H$ where $f(\underline{n}) = n'$. Moreover in this case we have $\sigma = \sigma_f$.* □

4 Algebraic term graph rewriting

We introduce now term graph rewriting according to the algebraic, double-pushout approach [Ehr87]. We prefer this approach (with respect to the more widely used "operational approach" introduced in [BvEG+87]) for two main reasons. Firstly, the actual result of a term reduction differs in the two approaches only for the case of circular redexes, and the interpretation proposed here for such reductions is consistent with the result of the algebraic approach. Secondly, the algebraic approach has been defined for an arbitrary category of graphs, and it can be applied to term graphs as a specific instance. Thus general theorems proposed in the literature and concerning parallel reductions, equivalences of reduction sequences [Kre87], and truly concurrent semantics in terms of event structures [CEL+96] can be applied to TGR as well.

Let **Graph** be a fixed category of graphs (below we will apply the general definitions to the category of term graphs defined in Section 3). The basic categorical construction in the algebraic definition of graph rewriting is that of pushout.

Definition 12 (pushout [ML71] and pushout complement [Ehr87]). Given a category C and two arrows $b : K \to B$, $d : K \to D$ of C, a triple $\langle H, h : B \to H, c : D \to H\rangle$ as in Figure 3 (a) is called a **pushout** of $\langle b, d\rangle$ if *[Commutativity]* $h \circ b = c \circ d$, and *[Universal Property]* for all objects H' and arrows $h' : B \to H'$ and $c' : D \to H'$, with $h' \circ b = c' \circ d$, there exists a unique arrow $f : H \to H'$ such that $f \circ h = h'$ and $f \circ c = c'$.

In this situation, H is called a **pushout object** of $\langle b, d\rangle$. Moreover, given arrows $b : K \to B$ and $h : B \to H$, a **pushout complement** of $\langle b, h\rangle$ is a triple $\langle D, d : K \to D, c : D \to H\rangle$ such that $\langle H, h, c\rangle$ is a pushout of b and d. In this case D is called a **pushout complement object** of $\langle b, h\rangle$. □

$$\begin{tikzcd}
K \arrow[r, "b"] \arrow[d, "d"'] & B \arrow[d, "h"] \arrow[ddr, "h'"] & \\
D \arrow[r, "c"] \arrow[drr, "c'"'] & H \arrow[dr, "f"] & \\
& & H'
\end{tikzcd}$$

(a)

$$\begin{tikzcd}
p = (L \arrow[d, "g"'] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, "k"] & R) \arrow[d, "h"] \\
G & D \arrow[l, "d"] \arrow[r, "b"'] & H
\end{tikzcd}$$

(b)

Fig. 3. (**a**) Pushout diagram. (**b**) Direct derivation as double-pushout construction.

Definition 13 (graph grammars, direct derivations [Ehr87]). A **(graph) production** $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$ is a pair of injective graph morphisms $l : K \rightarrow L$ and $r : K \rightarrow R$. The graphs L, K, and R are called the **left-hand side**, the **interface**, and the **right-hand side** of p, respectively. A **graph grammar** $\mathcal{G} = \{p_i\}_{i \in I}$ is a set of graph productions.[6]

Given a graph G, a graph production $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$, and an **occurrence** (i.e., a graph morphism) $g : L \rightarrow G$, a **direct derivation** α **from** G **to** H **using** p **(based on** g**)** exists if and only if the diagram in Figure 3 (b) can be constructed, where both squares are required to be pushouts in **Graph**. In this case, D is called the **context** graph, and we write $\alpha : G \Rightarrow_{p,g} H$, or simply $\alpha : G \Rightarrow_p H$. $\square$

In a graph-theoretical setting, the pushout object H of Figure 3 (a) can be understood as the gluing of graphs B and D, obtained by identifying the images of K along b and d. Therefore the double-pushout construction can be interpreted as follows. In order to apply the production p to G, we first need to find an occurrence of its left-hand side L in G, i.e., a graph morphism $g : L \rightarrow G$. Next, to model the deletion of that occurrence from G, we have to find a graph D and morphisms k and d such that the resulting square is a pushout: The context graph D is characterized categorically as the pushout complement object of $\langle l, g \rangle$. Finally, we have to embed the right-hand side R into D: This embedding is expressed by the right pushout.

The conditions for the existence of pushouts and of pushout complements depend on the category **Graphs** for which the above definitions are introduced. Since we are interested just in the graph productions that represent term rewrite rules, in the rest of the paper we shall consider such definitions in the category

[6] Usually the definition of a grammar also includes a "start" graph and a set of "terminal" graphs, specified in some way. A set of productions could be called, more appropriately, a *graph rewriting system*. We call it "grammar" to be consistent with the literature [Ehr87].

TGraph$_\Sigma$, and we will present conditions for the existence of pushouts and pushout complements (see Proposition 16) only for a specific format of productions, called *evaluation rules* (according to the name in [HP91], where the acyclic case is considered). Such evaluation rules are term graph productions satisfying some additional requirements that make them suitable to represent term rules. They are quite different from the term graph rewrite rules as defined in [BvEG$^+$87, KKSdV94].

Definition 14 (evaluation rules). An **evaluation rule** is a term graph production $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$ such that

1. L is a tree and it is not a single empty node. Let $\underline{n}$ be the root of L.
2. $K = (N_L, s_L \downarrow (N_L \backslash \{\underline{n}\}), s_L \downarrow (N_L \backslash \{\underline{n}\}))$, that is, K is obtained from L by making the successor and labeling functions undefined on the root. Morphism $l : K \to L$ is the inclusion; notice that l is an isomorphism on nodes and that $var(K) = var(L) \cup \{\underline{n}\}$.
3. If restricted to $var(L)$, morphism $r : K \to R$ is an isomorphism between $var(L)$ and $var(R)$. □

Let us make explicit the relationship between evaluation rules and rewrite rules. From every evaluation rule p one can easily unravel a term rewrite rule $\mathcal{U}[p]$; furthermore, for each rewrite rule R we propose a suitable representation as evaluation rule, $\mathcal{G}[R]$.

Definition 15 (from evaluation to rewrite rules and backwards). Let $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$ be an evaluation rule. The **unraveling** of p is the term rewrite rule $\mathcal{U}[p] : t \to s$ defined as follows.

1. $t = \mathcal{U}_L[\underline{n}]$ (where $\underline{n}$ is the root of L, as usual);
2. $s = \mathcal{U}_R[r(\underline{n})]\sigma$, where $\sigma : var(R) \to var(L)$ is the substitution defined as $\sigma(x) = y \quad \text{if } r(y) = x$.

We shall say that an evaluation rule p is **non-self-overlapping** if so is the rewrite rule $\mathcal{U}[p]$. A **term graph rewriting system** (shortly **TGRS**) $\mathcal{P}$ is a finite set of evaluation rules, $\mathcal{P} = \{p_i\}_{i \in I}$; $\mathcal{P}$ is called **orthogonal** if so is the term rewriting system $\mathcal{U}[\mathcal{P}] \stackrel{def}{=} \{\mathcal{U}[p] \mid p \in \mathcal{P}\}$.

The other way around, let $R : t \to s$ be a rational, left-finite and left-linear term rewrite rule, and let $t = f(t_1, \ldots, t_k)$ (t must have this form because it cannot be a variable). Then its **graph representation** $\mathcal{G}[R]$ is the production $\mathcal{G}[R] = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$, where

1. $L = \mathcal{G}[\overline{\{t\}}]$ (by $\overline{T}$ we denote the closure of T with respect to the subterm relation).
2. K and $l : K \to L$ are defined according to point 2 of Definition 14.
3. $R = \mathcal{G}[\overline{\{s, t_1, \ldots, t_k\}}]$.

4. $r : K \to R$ is defined as $r(\underline{n}) = s$, and $r(n) = \mathcal{U}_K[n]$ if $n \in N_K \backslash \{\underline{n}\}$ (this is well-defined because the nodes of R are subterms of $\{s, t_1, \ldots, t_k\}$, according to Definition 10). □

By the properties of the unraveling function and of evaluation rules (Definitions 10 and 14) it is routine to check that for each evaluation rule p the term rewrite rule $\mathcal{U}[p]$ is well-defined, rational, total, left-finite, and left-linear (the last two because the left-hand side of an evaluation rule is a tree). The substitution σ applied to $\mathcal{U}_R[r(\underline{n})]$ in the above definition is needed to ensure that $var(s) \subseteq var(t)$. Similarly, it follows directly from the definitions that $\mathcal{G}[R]$ is a well-defined evaluation rule for each term rewrite rule R satisfying the required conditions.

The next proposition ensures that if we consider a non-self-overlapping evaluation rule, then the existence of an occurrence morphism from its left-hand side to a term graph is a sufficient condition for the applicability of the rule, i.e., the pushout complement and the pushout of Figure 3 (b) can always be constructed.

Proposition 16 (existence of pushout complements and pushouts). *Let $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\to} R)$ be an evaluation which is not self-overlapping,*[7] *and let $g : L \to G$ be an occurrence morphism. Then in category* **TGraph** *there exists a pushout complement $\langle D, k : K \to D, d : D \to G\rangle$ of $\langle l, g\rangle$, where D is obtained from graph G by making the labeling and the successor function undefined on the image of the root of L.*

Moreover, a pushout $\langle H, h : R \to H, b : D \to H\rangle$ of the resulting arrow $k : K \to D$ and r always exists, and therefore in the above hypotheses there exists a direct derivation $G \Rightarrow_{p,g} H$. □

The proof of the last proposition is reported (for the equivalent category **Jungle**) in [CR93], where also general conditions for the existence of pushouts are presented.[8] It is worth stressing that in the hypotheses of the last proposition, the pushout complement object D has the same nodes of G; thus the nodes of G can be "traced" after the rewriting. More formally, there is a total function, called the **track function** [HP91] $tr : N_G \to N_H$, defined as $tr(n) = b(n)$ for all $n \in N_G$ $(= N_D)$.

5 Adequacy of algebraic term graph rewriting for rational parallel term rewriting

The relationship between term and term graph rewriting has been nicely formalized in [KKSdV94] with the notion of *adequate mapping* between rewriting

[7] Without this condition the statement would not be true, because the *identification condition* [Ehr87] may not be satisfied.

[8] An interesting fact, which is not relevant for this paper, is that the pushout of two arrows exists in $\mathbf{TGraph}_\Sigma$ iff the associated substitutions unify, and in this case the pushout is a most general unifier.

systems. We recall here the definition, referring to that paper for the precise motivations. A *rewriting system* is defined in this context as a triple (A, R, S), where A is a set of states (in our case terms or graphs), R is a set of rules, and S is a set of reduction sequences, closed under certain operations.

Definition 17 (adequate mapping between rewriting systems). Let (A_1, R_1, S_1) and (A_2, R_2, S_2) be two rewriting systems. A mapping $\mathcal{U} : A_1 \to A_2$ is **adequate** if:

[*Surjectivity*] $\mathcal{U}$ is surjective;
[*Preservation of normal forms*] $a \in A_1$ is in normal form iff $\mathcal{U}[a] \in A_2$ is in normal form.[9]
[*Preservation of reductions*] If $a \to^* a'$ with a reduction sequence in S_1, then $\mathcal{U}[a] \to^* \mathcal{U}[a']$ with a reduction sequence in S_2.[10]
[*Cofinality*] For $a \in A_1$ and $b \in A_2$, if $\mathcal{U}[a] \to^* b$ in S_2, then there is a a' in A_1 such that $a \to^* a'$ in S_1 and $b \to^* \mathcal{U}[a']$ in S_2. □

We show below that the unraveling function $\mathcal{U}$ introduced in the previous sections is an adequate mapping from a given orthogonal TGRS $\mathcal{P}$ to its unraveled orthogonal TRS $\mathcal{U}[\mathcal{P}]$, restricting the allowed parallel reduction sequences to the rational ones.

Definition 18 (rational parallel reduction sequences). Let $\mathcal{R}$ be an orthogonal TGR. A parallel reduction $t \to_{\Phi} t'$ is **rational** if the term obtained by marking in t all the occurrences of redexes of Φ is rational. A **rational parallel reduction sequence** is a parallel reduction sequence where each step is rational. A **rational** TRS is a triple $(CT_{\Sigma}^{rat}, \mathcal{R}, \mathcal{S}_{rat}(\mathcal{R}))$, where CT_{Σ}^{rat} is the set of all rational terms in CT_{Σ}, $\mathcal{R}$ is an orthogonal TRS where all right-hand sides are rational terms, and $\mathcal{S}_{rat}(\mathcal{R})$ is the set of all rational parallel reduction sequences using rules in $\mathcal{R}$. □

It is worth stressing that strong confluence and confluence (Proposition 7) also hold for rational parallel reductions.

In order to prove the adequacy of the unraveling function, we need the following fundamental result, which formalizes the intuition presented in the introduction: A single (possibly cyclic) term graph reduction can be interpreted as a parallel (possibly infinite) term reduction.

Theorem 19 (from graph reductions to parallel term reductions). *Let $p = (L \xleftarrow{l} K \xrightarrow{r} R)$ be a non-self-overlapping evaluation rule, let G be a term graph, and let $g : L \to G$ be an occurrence morphism. By Proposition 16 we know that $G \Rightarrow_{p,g} H$, with the corresponding track function $tr : N_G \to N_H$. Then for each node $n \in N_G$, we have*

$$\mathcal{U}_G[n] \to_{\Phi} \mathcal{U}_H[tr(n)]\sigma$$

[9] A state is in normal form if there is no reduction sequence starting from it.
[10] We consider only reduction sequences of finite length.

where Φ is the (possibly infinite) parallel redex $\Phi = \{(\mathcal{O}(\pi), \mathcal{U}[p]) \mid \pi$ is a path in G from n to $g(\underline{n})\}$ ($\underline{n}$ is the root of L), and where substitution $\sigma : var(H) \to var(G)$ is defined as $\sigma(x) = y$ if $y \in var(G) \wedge tr(y) = x$, and $\sigma(x) = \perp$ if $\nexists y \in var(G) \,.\, tr(y) = x$.

Proof. For the sake of simplicity, let us assume that $var(G) = \emptyset$, which implies that substitution σ becomes $x\sigma = \perp$ for all $x \in var(H)$.[11]

Let $R : l_p \to r_p$ be the rewrite rule $\mathcal{U}[p]$. We first have to show that Φ is a parallel redex of $\mathcal{U}_G[n]$, i.e., that for each $(\mathcal{O}(\pi), R) \in \Phi$, there is a substitution τ such that $\mathcal{U}_G[n]/\mathcal{O}(\pi) = l_p\tau$. In fact we have

$$\begin{array}{ll} \mathcal{U}_G[n]/\mathcal{O}(\pi) & = [\text{because } \pi \text{ is a path from } n \text{ to } g(\underline{n}) \text{ (see Definition 10)}] \\ \mathcal{U}_G[g(\underline{n})] & = [\text{by point 1 of Proposition 11 and by Definition 15}] \\ \mathcal{U}_L[\underline{n}]\sigma_g = l_p\sigma_g & \end{array}$$

Thus all redexes in Φ are realized by the same substitution σ_g. Let W be the set of all occurrences of paths from n to $g(\underline{n})$ in G (thus W is the set of all occurrences of redexes in Φ), and let $\langle w_1, w_2, \ldots \rangle$ be an arbitrary but fixed enumeration of W such that if $w_i < w_j$, then $i < j$ (in words, no occurrence can be followed by a prefix). For all $i < \omega$, define

$$t_i(u) = \begin{cases} \mathcal{U}_G[n](u) & \text{if } \forall j > i \,.\, u \not\geq w_j \\ \perp & \text{otherwise.} \end{cases}$$

Obviously, $\{t_i\}_{i<\omega}$ is a chain and $\bigcup_{i<\omega}\{t_i\} = \mathcal{U}_G[n]$. Moreover, (w_j, R) is a redex of t_i iff $j \leq i$: This follows from orthogonality, and observing that $t_i(w_j) \neq \perp$ iff $j \leq i$. Therefore, according to Definition 5, the subset Φ_i of Φ including all redexes of t_i is $\Phi_i = \{(w_1, R), \ldots, (w_i, R)\}$ (in particular, $\Phi_0 = \emptyset$). Thus by Definition 5 we have $\mathcal{U}_G[n] \to_\Phi \bigcup_{i<\omega}\{s_i\}$, where, for each i, $t_i \to_{\Phi_i} s_i$.

It remains to prove that $\bigcup_{i<\omega}\{s_i\} = \mathcal{U}_H[tr(n)]\sigma$. Let us first show by induction that $\forall i < \omega \,.\, s_i \leq \mathcal{U}_H[tr(n)]\sigma$.[12]

[Base Case] Since $\Phi_0 = \emptyset$, we have that $s_0 = t_0$. Let $u \in \mathcal{O}(s_0)$. By definition $u \not\geq w$ for all $w \in W$. Thus we have

$$\begin{array}{rl} s_0(u) = \mathcal{U}_G[n](u) = & [\text{assuming that } \pi \text{ is the only path from } n \text{ to } n' \text{ in} \\ & \;G \text{ with occurrence } u, \text{ and since } var(G) = \emptyset] \\ l_G(n') = & [\text{by the explicit definition of } D \text{ (Proposition 16),} \\ & \;\text{since } n' \neq g(\underline{n})] \\ l_D(n') = & [\text{by properties of morphisms and definition of } tr] \\ l_H(b(n')) = l_H(tr(n')) = & [\text{because the track function preserves all paths (like} \\ & \;\pi) \text{ not containing } g(\underline{n})] \\ \mathcal{U}_H[tr(n)](u) = & [\sigma \text{ does not affect occurrences of operators, like } u] \\ \mathcal{U}_H[tr(n)]\sigma(u) & \end{array}$$

[11] The general case needs an additional technical lemma showing that the trace function maps variables to variables in an injective way (thus σ is well-defined); this can be proved by a careful inspection of the double-pushout diagram.

[12] It is worth recalling that $t \leq t' \Leftrightarrow \forall u \in \mathcal{O}(t) \,.\, t(u) = t'(u)$.

[Inductive Case] We must show that $s_i \leq \mathcal{U}_H[tr(n)]\sigma \Rightarrow s_{i+1} \leq \mathcal{U}_H[tr(n)]\sigma$. By the above definitions, the only redex of t_{i+1} which is not of t_i is $\Delta_{i+1} = (w_{i+1}, R)$. If $t_{i+1} \to_{\Phi_i} t'$, let $\Delta_{i+1} \backslash \Phi_i$ be the residual, and let $V = \{v_1, \ldots, v_k\}$ its set of occurrences. Then, if τ is the substitution such that $l_p\tau = t_{i+1}/v$ for all $v \in V$,[13] a careful inspection reveals that s_{i+i} can be defined in term of s_i as

$$s_{i+1}(u) = \begin{cases} s_i(u) & \text{if } u \in \mathcal{O}(s_i) \\ r_p\tau(w) & \text{if there is a } v \in V \text{ such that } u = vw, \text{ and} \\ & w \in \mathcal{O}(r_p\tau) \\ \perp & \text{otherwise.} \end{cases}$$

Exploiting the induction hypothesis, it remains to show that $s_{i+1}(vw) = \mathcal{U}_H[tr(n)]\sigma(vw)$ only for $v \in V$ and $w \in \mathcal{O}(r_p\tau)$. This can be done by combining the techniques used for the acyclic case in [HP91, CR93], and those used for the Base Case above, because the definition of t_{i+1} ensures that no redex of Φ appears in substitution τ (although τ may substitute an infinite term for a variable).

Finally it remains to prove that $\mathcal{U}_H[tr(n)]\sigma \leq \bigcup_{i<\omega}\{s_i\}$, and this can be done as follows. If rule R is not collapsing, it can be shown that $\bigcup_{i<\omega}\{s_i\}$ is a total term (and therefore it is a maximal element of the approximation ordering), because the least depth of $\perp$'s in s_i tends to infinity. If instead R is collapsing, then it is easy to check that $s_0 = s_1 = \ldots = \bigcup_{i<\omega}\{s_i\}$, and that $\mathcal{U}_H[tr(n)]\sigma = s_0$ using the Base Case above and the fact that the only variable in H, $tr(g(\underline{n}))$, is substituted for $\perp$ by σ. □

Exploiting this main result, we can prove the adequacy of the unraveling function. One minor problem is due to the fact that the unraveling of a term graph is not a term, but a set of terms. Thus either we consider TRS where a state can be a set of terms (which seems quite unnatural), or we have to consider *pointed* term graphs, i.e., graphs with a distinguished node, and to assume that the $\mathcal{U}$ unravels each graph at that specific node. However, in this last case, unraveling would not preserve normal forms, because a graph may contain a redex in a part that is not reachable from the distinguished node (in the *garbage*), and such redex would not appear in the unraveled term. Full adequacy could be recovered by adding garbage collection at each graph rewriting step (as it is done in [BvEG+87, KKSdV94]) but we prefer to prove a slightly weaker result without modifying the graph rewriting formalism.

Theorem 20 (adequacy of TGR for rational parallel TR). *Given an orthogonal TGRS $\mathcal{P}$, let $(\mathcal{G}, \mathcal{P}, \mathcal{S}(\mathcal{P}))$ be the rewriting system where $\mathcal{G}$ is the set of all pointed term graphs over Σ and $\mathcal{S}(\mathcal{P})$ is the set of all graph reduction sequences using rules in $\mathcal{P}$ and such that the track function preserves the distinguished node.*

[13] It can be checked that all redexes in $\Delta_{i+1} \backslash \Phi_i$ are realized by the same substitution in t'.

Then the unraveling function $\mathcal{U}$ is a mapping from $(\mathcal{G}, \mathcal{P}, \mathcal{S}(\mathcal{P}))$ to the rational TRS $(CT_{\Sigma}^{rat}, \mathcal{U}[\mathcal{P}], \mathcal{S}_{rat}(\mathcal{U}[\mathcal{P}]))$ which satisfies all the conditions of adequate mappings (Definition 17), except of the preservation of normal forms. Nevertheless, it satisfies the following weaker version: [Weak preservation of normal forms] *if $a \in A_1$ is a normal form, then so is $\mathcal{U}[a] \in A_2$.*

Proof. According to Definition 17, we have:

[*Surjectivity*] Immediate by Definition 10, using pointed graphs.

[*Weak preservation of normal forms*] Follows directly from point 2 of Proposition 11: if there is a redex of rule $\mathcal{U}[p]$ in term $\mathcal{U}[G]$, then there is an occurrence morphism from the left-hand side of p to G.

[*Preservation of reductions*] By repeated applications of Theorem 19.

[*Cofinality*] Let G be a pointed graph, and suppose that $\mathcal{U}[G] \to^* t$ via a rational reduction sequence. We show by induction on the length of the sequence that there is a G' such that $t \to^* \mathcal{U}[G']$, and $G \Rightarrow^* G'$.

[Base Case] If $\mathcal{U}[G] \to_{\Phi} t$ and Φ is rational, it is possible to show that there exists a finite set M of occurrence morphisms of rules of $\mathcal{P}$ in G, such that their "unraveling" (that we do not define formally) $\mathcal{U}[M]$ is a parallel redex of $\mathcal{U}[G]$ containing Φ. Then let G' be the graph obtained by applying to G all the occurrence morphisms in M in any order (by orthogonality G' is well-defined). Then it can be shown that $t \to_{\mathcal{U}[M] \backslash \Phi} \mathcal{U}[G']$.

$$\begin{tikzcd}
\mathcal{U}[G] \arrow[r, "n"] & t' \arrow[r, "*", "d"'] \arrow[d, "\Phi"'] & \mathcal{U}[G'] \arrow[d, "\Phi\backslash d"] & \\
& t \arrow[r, "*", "d\backslash\Phi"'] & t'' \arrow[r, "*"] & \mathcal{U}[G'']
\end{tikzcd}$$

$$\begin{tikzcd}
G \arrow[rr, Rightarrow, "*"] & & G' \arrow[r, Rightarrow, "*"] & G''
\end{tikzcd}$$

Fig. 4. Proof of cofinality.

[Inductive Case] Suppose that $\mathcal{U}[G] \to^n t' \to_{\Phi} t$, and consider the diagram in Figure 5. By inductive hypothesis there exists G' such that $G \Rightarrow^* G'$ and $t' \to^* \mathcal{U}[G']$. By strong confluence of rational parallel reductions, there exists a t'' such that the square commutes, and the residual $\Phi\backslash d$ is a rational parallel redex. Then applying the Base Case to $\mathcal{U}[G'] \to_{\Phi\backslash d} t''$, we have that there exists a G'' such that $G' \Rightarrow^* G''$ and $t'' \to^* \mathcal{U}[G'']$. □

6 Conclusions

We showed that by exploiting the complete partial ordered structure of (infinite, partial) terms, a notion of infinite parallel reduction can be defined. Moreover, we proved that this notion can be used to relate Term Graph Rewriting with cycles and Term Rewriting, formalizing the intuition that a single graph reduction correspond to a (possibly infinite) *parallel* term reduction. This result has been used to show that cyclic graph rewriting is adequate for rational term rewriting, exploiting the notion of adequacy proposed in [KKSdV94].

In our opinion, our adequacy result is more satisfactory than the corresponding results in [KKSdV94] which relates graph reductions with infinitary term reductions for two reasons. Firstly, unraveling preserves the length of reductions sequences, which is a pleasant property. Secondly, collapsing rules and circular redexes do not constitute in our framework technical problems, as it is the case in the other approach, but are treated in a uniform way. A deeper analysis of these aspects will be included in the full version of the paper.

References

[AN80] A. Arnold and M. Nivat. The metric space of infinite trees. algebraic and topological properties. *Fundamenta Informaticae*, 4:445–476, 1980.

[BvEG+87] H.P. Barendregt, M.C.J.D. van Eekelen, J.R.W. Glauert, J.R. Kennaway, M.J. Plasmeijer, and M.R. Sleep. Term graph reduction. In *Proceedings PARLE*, volume 259 of *LNCS*, pages 141–158. Springer Verlag, 1987.

[CEL+96] A. Corradini, H. Ehrig, M. Löwe, U. Montanari, and F. Rossi. An Event Structure Semantics for Graph Grammars with Parallel Productions. In J. Cuny, H. Ehrig, G. Engels, and G. Rozenberg, editors, *Proceedings of the 5th International Workshop on Graph Grammars and their Application to Computer Science*, volume 1073 of *LNCS*. Springer Verlag, 1996.

[CMR+91] A. Corradini, U. Montanari, F. Rossi, H. Ehrig, and M. Löwe. Logic programming and graph grammars. In H. Ehrig, H.-J. Kreowski, and G. Rozenberg, editors, *Proceedings of the 4th International Workshop on Graph-Grammars and Their Application to Computer Science*, volume 532 of *LNCS*, pages 221–237. Springer Verlag, 1991.

[Cor93] A. Corradini. Term Rewriting in CT_Σ. In *Proceedings CAAP '93*, volume 668 of *LNCS*, pages 468–484. Springer Verlag, 1993.

[CR93] A. Corradini and F. Rossi. Hyperedge Replacement Jungle Rewriting for Term Rewriting Systems and Logic Programming. *Theoret. Comput. Sci.*, 109:7–48, 1993.

[DK89] N. Dershowitz and S. Kaplan. Rewrite, rewrite, rewrite, rewrite, rewrite.... In *Proc. POPL'89, Austin*, pages 250–259, 1989.

[DKP89] N. Dershowitz, S. Kaplan, and D.A. Plaisted. Infinite Normal Forms (plus corrigendum). In *Proc. ICALP'89*, pages 249–262, 1989.

[Ehr87] H. Ehrig. Tutorial introduction to the algebraic approach of graph-grammars. In H. Ehrig, M. Nagl, G. Rozenberg, and A. Rosenfeld, editors, *Proceedings of the 3rd International Workshop on Graph-Grammars and Their Application to Computer Science*, volume 291 of *LNCS*, pages 3–14. Springer Verlag, 1987.

[FRW90] W.M. Farmer, J.D. Ramsdell, and R.J. Watro. A correctness proof for combinator reduction with cycles. *ACM Trans. Program. Lang. Syst.*, 12:123–134, 1990.

[FW91] W.M. Farmer and R.J. Watro. Redex capturing in term graph rewriting. In R.V. Book, editor, *Proceedings of the 4th International Conference on Rewriting Techniques and Applications (RTA'91)*, volume 488 of *LNCS*, pages 13–24. Springer Verlag, 1991.

[GTWW77] J.A. Goguen, J.W. Tatcher, E.G. Wagner, and J.R Wright. Initial Algebra Semantics and Continuous Algebras. *Journal of the ACM*, 24:68–95, 1977.

[HL91] G. Huet and J-J. Lévy. Computations in Orthogonal Rewriting Systems, I. In J.-L. Lassez and G. Plotkin, editors, *Computational Logic: Essays in honour of J. A. Robinson*, pages 395–414. MIT Press, 1991.

[HP91] B. Hoffmann and D. Plump. Implementing Term Rewriting by Jungle Evaluation. *Informatique théorique et Applications/Theoretical Informatics and Applications*, 25:445–472, 1991.

[Ken91] J.R. Kennaway. Graph rewriting in some categories of partial morphisms. In *Graph Grammars and their Application in Computer Science*, volume 532 of *LNCS*, pages 490–504. Springer Verlag, 1991.

[KKSdV94] J.R. Kennaway, J.W. Klop, M.R. Sleep, and F.J. de Vries. On the Adequacy of Graph Rrewriting for Simulating Term Rewriting. *ACM Trans. Program. Lang. Syst.*, 16:493–523, 1994.

[Kre87] H.-J. Kreowski. Is parallelism already concurrency? Part 1: Derivations in graph grammars. In H. Ehrig, M. Nagl, G. Rozenberg, and A. Rosenfeld, editors, *Proceedings of the 3rd International Workshop on Graph-Grammars and Their Application to Computer Science*, volume 291 of *LNCS*, pages 343–360. Springer Verlag, 1987.

[ML71] S. Mac Lane. *Categories for the working mathematician*. Springer Verlag, 1971.

[Sta80] J. Staples. Computation of graph-like expressions. *Theoret. Comput. Sci.*, 10:171–195, 1980.