Math. Struct. in Comp. Science (2018), *vol.* 28, *pp.* 1363–1414. © Cambridge University Press 2018
doi:10.1017/S0960129518000166

Convergence in infinitary term graph rewriting systems is simple

PATRICK BAHR

Computer Science Department, IT University of Copenhagen
2300 Copenhagen, Denmark
Email: paba@itu.dk

Received 1 February 2012; revised 1 November 2017

Term graph rewriting provides a formalism for implementing term rewriting in an efficient manner by emulating duplication via sharing. Infinitary term rewriting has been introduced to study infinite term reduction sequences. Such infinite reductions can be used to model *non-strict evaluation*. In this paper, we unify term graph rewriting and infinitary term rewriting thereby addressing both components of *lazy evaluation*: non-strictness and sharing. In contrast to previous attempts to formalise infinitary term graph rewriting, our approach is based on a simple and natural generalisation of the modes of convergence of infinitary term rewriting. We show that this new approach is better suited for infinitary term graph rewriting as it is simpler and more general. The latter is demonstrated by the fact that our notions of convergence give rise to two independent canonical and exhaustive constructions of infinite term graphs from finite term graphs via metric and ideal completion. In addition, we show that our notions of convergence on term graphs are sound w.r.t. the ones employed in infinitary term rewriting in the sense that convergence is preserved by unravelling term graphs to terms. Moreover, the resulting infinitary term graph calculi provide a unified framework for both infinitary term rewriting and term graph rewriting, which makes it possible to study the correspondences between these two worlds more closely.

1. Introduction

Term graphs are a generalisation of terms, which allow us to avoid duplication of subterms and instead use pointers in order to refer to the same subterm several times. In this paper, we aim to extend the theory of infinitary term rewriting to the setting of term graphs.

As the basis for our infinitary calculi, we use the well-established term graph rewriting formalism of Barendregt et al. (1987) as it will allow us to draw on the work investigating the relation between (infinitary) term rewriting on the one hand and term graph rewriting on the other hand (Kennaway et al. 1994).

In order to devise an infinitary calculus, we have to conceive a notion of convergence that constrains reductions of transfinite length in a meaningful way. To this end, we generalise the metric on terms that is used to define convergence for infinitary term rewriting (Dershowitz et al. 1991) to term graphs. In a similar way, we generalise the partial order on terms that has been recently used to define a closely related notion of convergence for infinitary term rewriting (Bahr 2014). The use of two different – but on terms closely related – approaches to convergence will allow us both to assess

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

the appropriateness of the resulting infinitary calculi and to compare them against the corresponding infinitary calculi of term rewriting.

The focus of the present work is primarily on the foundational aspects of infinitary term graph rewriting. That is, our major concerns are the underlying notions of convergence and their appropriateness. That is why we only consider weak forms of convergence, i.e. notions of convergence that are purely based on the convergence of the terms or term graphs along a reduction, as opposed to strong convergence (Kennaway et al. 1995) that also considers the positions of contracted redexes.

1.1. *Motivation*

1.1.1. *Lazy evaluation.* Many functional programming languages allow deferring the evaluation of an expression until its 'value' is needed for evaluation in the context that it appears. The functional programming language Haskell (Marlow 2010) has this semantics by default. But other languages typically provide means to selectively enable this evaluation strategy as well, e.g. F#.

As an example what this so-called *non-strict evaluation* semantics offers, consider the following function definition for `from`:

```
from(n) = n :: from(s(n))
```

Here, we use the binary infix symbol `::` to denote the list constructor cons and `s` for the successor function on natural numbers. Intuitively, `from` constructs for each number n the infinite list of consecutive numbers starting from n.

Obviously, we cannot use the infinite list generated by `from` directly. By its very nature, viz. constructing an infinite list, the evaluation of an expression of the form `from n` cannot terminate. However, what we can do is use the expression `from n` in a context in which we only read a finite prefix of the infinite list conceptually defined by `from`.

Non-strict evaluation enables this program construction as it delays the evaluation of a subexpression until its result is actually required for further evaluation of the expression. Note that this non-strict semantics is not only a conceptual elegancy but in fact one of the major features that make functional programs highly modular (Hughes 1989).

Term rewriting can be used to study functional programs. A functional program essentially consists of functions defined by a set of equations and an expression that is supposed to be evaluated according to these equations. The conceptual process of evaluating an expression is nothing else than term rewriting.

The above definition of the function `from` is represented as a term rewriting system with the following rule:

$$from(x) \to x :: from(s(x))$$

Starting with the term $from(0)$, we thus obtain the following infinite reduction:

$$from(0) \to 0 :: from(s(0)) \to 0 :: s(0) :: from(s(s(0))) \to \dots$$

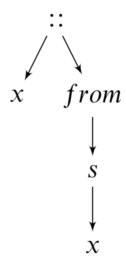
Infinitary term rewriting (Kennaway and de Vries 2003) provides a notion of convergence that may assign a meaningful result term to such an infinite reduction provided there exists one. For instance, the above reduction converges to the infinite term

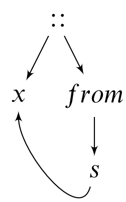
https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

$0 :: s(0) :: s(s(0)) :: \dots$, which represents the infinite list of numbers $0, 1, 2, \dots$. This extension of term rewriting with explicit limit constructions for non-terminating reductions allows us to directly reason about non-terminating functions and infinite data structures.

But, in a practical implementation, non-strict evaluation is rarely left to its own devices. Usually, non-strict evaluation is implemented as part of lazy evaluation (Henderson and Morris Jr. 1976), which complements a non-strict evaluation strategy with *sharing*. The latter avoids duplication of subexpressions and instead uses pointers. For example, the function `from` above duplicates its argument `n` – it occurs twice on the right-hand side of the defining equation. A lazy evaluator simulates this duplication by inserting two pointers pointing to the actual argument. Sharing is a natural companion for non-strict evaluation as it avoids re-evaluation of expressions that are duplicated before they are evaluated.

The underlying formalism that is typically used to obtain sharing for functional programming languages is term graph rewriting (Peyton-Jones 1987; Plasmeijer and van Eekelen 1993). Term graph rewriting (Barendregt et al. 1987; Plump 1999) uses graphs to represent terms, thus allowing multiple arcs to point to the same node. For example, the right-hand side $x :: from(s(x))$ of the term rewrite rule defining the function $from$ can be represented as a term graph in the following two ways:

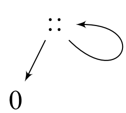




The former is a tree and corresponds directly to the original term representation; the variable x still occurs twice. The latter is a proper graph (i.e. not a tree) and only contains one occurrence of the variable x. The two original occurrences of the variable are represented by two ingoing edges.

The non-strictness part of lazy evaluation is covered by infinitary rewriting, whereas term graphs give us a model for sharing. We aim to unify the two formalisms into one calculus by endowing term graph rewriting with a notion of convergence. This unification will allow us to model both aspects of lazy evaluation within the same calculus.

1.1.2. *Rational terms.* In their full generality, term graphs can do more than only share common subexpressions. Through *cycles* term graphs may also provide a finite representation of certain infinite terms – so-called *rational terms*. For example, the infinite term $0 :: 0 :: 0 :: \dots$ can be represented as the finite term graph



https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

A single node on a cycle in a term graph represents infinitely many corresponding subterms. Accordingly, the contraction of a single term graph redex may not correspond to only a single term rewrite step but instead a transfinite term reduction that contracts infinitely many term redexes.

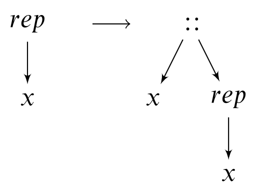
For example, if we apply the rewrite rule $0 \to s(0)$ to the above term graph, we obtain a term graph that represents the term $s(0) :: s(0) :: s(0) :: \ldots$, which can only be obtained from the term $0 :: 0 :: 0 :: \ldots$ via a *transfinite* term reduction with the rule $0 \to s(0)$. Kennaway et al. (1994) investigated this correspondence between cyclic term graph rewriting and infinitary term rewriting. They were able to characterise a subset of transfinite term reductions – called *rational reductions* – that can be simulated by a corresponding finite term graph reduction. The above reduction from the term $0 :: 0 :: 0 :: \ldots$ is an example of such a rational reduction.

With the help of a unified formalism for infinitary and term graph rewriting, it should be easier to study the correspondence between infinitary term rewriting and finitary term graph rewriting further. The move from an infinitary term rewriting system to a term graph rewriting system is then only a change in the degree of sharing if we use infinitary term graph rewriting as a common framework.

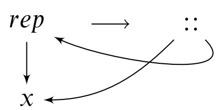
For example, consider the term rewrite rule $rep(x) \to x :: rep(x)$, which defines a function *rep* that repeats its argument infinitely often:

$$rep(0) \to 0 :: rep(0) \to 0 :: 0 :: rep(0) \to 0 :: 0 :: 0 :: rep(0) \to \quad \ldots \quad 0 :: 0 :: 0 :: \ldots$$

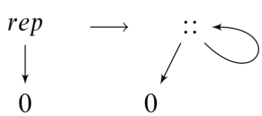
This reduction happens to be not a rational reduction in the sense of Kennaway et al. (1994). The move from the term rule $rep(x) \to x :: rep(x)$ to a term graph rule is a simple matter of introducing sharing of common subexpressions:



is represented by



Instead of creating a fresh copy of the redex on the right-hand side, the redex is reused by placing an edge from the right-hand side of the rule to its left-hand side. This rewrite rule allows us to represent the infinite reduction approximating the infinite term $0 :: 0 :: 0 :: \ldots$ with the following single step term graph reduction:



By its cyclic structure, the resulting term graph represents the infinite term $0 :: 0 :: 0 :: \ldots$.

Since both transfinite term reductions and the corresponding finite term graph reductions can be treated within the same formalism, we hope to provide a tool for studying the ability of cyclic term graph rewriting to finitely represent transfinite term reductions.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

1.2. *Contributions and related work*

1.2.1. *Contributions.* The main contributions of this paper are as follows:

(i) We devise a simple partial order on term graphs based on graph homomorphisms. We show that this partial order forms a complete semi-lattice and thus is technically suitable for defining a notion of convergence.
(ii) We devise a simple metric on term graphs and show that it forms a complete ultrametric space on term graphs.
(iii) Based on the partial order and the metric we define a notion of weak convergence for infinitary term graph rewriting. We show that the partial order convergence subsumes the metric convergence.
(iv) We confirm that the partial order and the metric on term graphs generalise the partial order and the metric that is used for infinitary term rewriting. Moreover, we show that the corresponding notions of convergence are preserved by unravelling term graphs to terms thus establishing the soundness of our notions of convergence on term graphs w.r.t. the convergence on terms.
(v) Finally, we show that both the partial order and the metric provide completion constructions – ideal completion and metric completion, respectively – that construct the set of finite and infinite term graphs from the set of finite term graphs.

In this paper, we study the foundations of infinitary term graph rewriting and therefore focus purely on weak notions of convergence, i.e. notions that are based on the sequence of term graphs produced along a term graph reduction. Similar to infinitary term rewriting, weak notions of convergence for infinitary term graph rewriting are difficult to study and often manifest some unexpected behaviour. In particular, soundness and completeness properties w.r.t. infinitary term rewriting are hard to come by. Yet, we gathered much evidence that support the appropriateness of our infinitary calculi. More evidence can be found when moving to strong convergence, which does exhibit solid soundness and completeness properties w.r.t. infinitary term rewriting (Bahr 2012a).

1.2.2. *Related work.* Ariola and Klop (1997) recognised that adding explicit recursion mechanism in the form of *letrec* to the lambda calculus may break confluence. In order to reconcile this, Ariola and Blom (2002, 2005) developed a notion of skew confluence that allows them to define an infinite normal form in the vein of Böhm trees.

In previous work, we have investigated notions of convergence for term graph rewriting (Bahr 2012b). The approach that we have taken in that work is very similar to the approach adopted in this paper: By generalising the metric and the partial order on terms to term graphs, we devised a weak notion of convergence for infinitary term graph rewriting. However, both the metric and the partial order on term graphs are very carefully crafted in order to make them very similar to the corresponding structures on terms. While the thus obtained two notions of convergence manifest the same correspondence that is known from infinitary term rewriting (Bahr 2014), they are too restrictive as we will illustrate in this paper. Due to the close resemblance to the convergence on terms, these notions of convergence are not able to capture all forms of sharing appropriately.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

In this paper, we follow a different approach by taking the arguably simplest generalisation of the metric and the partial order to term graphs. We will show that this approach is better suited for infinitary term graph rewriting as it lifts the restrictions that we observe in our previous formalisation (Bahr 2012b).

1.3. *Overview*

The structure of this paper is as follows: In Section 2, we give an overview of infinitary term rewriting including the necessary background for metric spaces and partially ordered sets. Section 3 provides the necessary theory for graphs and term graphs. Sections 4 and 5 form the core of this paper. In these sections, we study the partial order and the metric on term graphs that are the basis for the notions of convergence we consider in this paper. In Section 6, we use these two notions of convergence to study two corresponding infinitary term graph rewriting calculi. Sections 7 and 8 are concerned with forms of soundness and completeness properties of our notions of convergence. In the former, we show that both notions of convergence generalise the corresponding notions of convergence on terms and that they are preserved under unravelling term graphs to terms. In the latter, we show that the set of (finite and infinite) term graphs arises both as the metric completion and the ideal completion of the set of finite term graphs.

2. Infinitary term rewriting

For devising an infinitary calculus, we have to devise a notion of convergence that constrains transfinite reductions in a meaningful way. Before pondering over the right approach to an infinitary calculus of term graph rewriting, we want to provide a brief overview of infinitary term rewriting (Kennaway and de Vries 2003; Bahr 2014). In this paper, we will only consider weak notions of convergence, i.e. convergence is solely determined by the sequence of terms or term graphs that are produced along a reduction (Dershowitz et al. 1991).

We assume the reader to be familiar with the basic theory of ordinal numbers, orders and topological spaces (Kelley 1955), as well as term rewriting (Terese 2003). In the following, we briefly recall the most important notions.

2.1. *Sequences*

We use the von Neumann definition of ordinal numbers. That is, an *ordinal number* (or simply *ordinal*) α is the set of all ordinal numbers strictly smaller than α. In particular, each natural number $n \in \mathbb{N}$ is an ordinal number with $n = \{0, 1, \dots, n-1\}$. The least infinite ordinal number is denoted by ω and is the set of all natural numbers. Ordinal numbers will be denoted by lower case Greek letters $\alpha, \beta, \gamma, \delta, \lambda, \iota$.

A *sequence* S of length α in a set A, written $(a_\iota)_{\iota<\alpha}$, is a function from α to A with $\iota \mapsto a_\iota$ for all $\iota \in \alpha$. We use $|S|$ to denote the length α of S. If α is a limit ordinal, then S is called *open*. Otherwise, it is called *closed*. If α is a finite ordinal, then S is called *finite*. Otherwise, it is called *infinite*. For a finite sequence $(a_i)_{i<n}$, we also use the notation

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

$\langle a_0, a_1, \ldots, a_{n-1}\rangle$. In particular, $\langle\rangle$ denotes the empty sequence. We write A^* for the set of all finite sequences in A.

The *concatenation* $(a_\iota)_{\iota<\alpha} \cdot (b_\iota)_{\iota<\beta}$ of two sequences $(a_\iota)_{\iota<\alpha}$ and $(b_\iota)_{\iota<\beta}$ is the sequence $(c_\iota)_{\iota<\alpha+\beta}$ with $c_\iota = a_\iota$ for $\iota < \alpha$ and $c_{\alpha+\iota} = b_\iota$ for $\iota < \beta$. A sequence S is a (proper) *prefix* of a sequence T, denoted $S \leqslant T$ (respectively $S < T$), if there is a (non-empty) sequence S' with $S \cdot S' = T$. The prefix of T of length $\beta \leqslant |T|$ is denoted $T|_\beta$. The thus defined binary prefix relation $\leqslant$ forms a complete semilattice (cf. Section 2.3). Similarly, a sequence S is a (proper) *suffix* of a sequence T if there is a (non-empty) sequence S' with $S' \cdot S = T$.

2.2. *Metric spaces*

Given a set M, a pair $(M, \mathbf{d})$ is called a *metric space* if $\mathbf{d}\colon M \times M \to \mathbb{R}_0^+$ is a function satisfying $\mathbf{d}(x,y) = 0$ iff $x = y$ (identity), $\mathbf{d}(x,y) = \mathbf{d}(y,x)$ (symmetry) and $\mathbf{d}(x,z) \leqslant \mathbf{d}(x,y) + \mathbf{d}(y,z)$ (triangle inequality), for all $x,y,z \in M$. If $\mathbf{d}$, instead of the triangle inequality, satisfies the stronger property $\mathbf{d}(x,z) \leqslant \max\{\mathbf{d}(x,y), \mathbf{d}(y,z)\}$ (strong triangle), then $(M, \mathbf{d})$ is called an *ultrametric space*. Let $(a_\iota)_{\iota<\alpha}$ be a sequence in a metric space $(M, \mathbf{d})$. The sequence $(a_\iota)_{\iota<\alpha}$ *converges* to an element $a \in M$, written $\lim_{\iota\to\alpha} a_\iota$, if, for each $\varepsilon \in \mathbb{R}^+$, there is a $\beta < \alpha$ such that $\mathbf{d}(a, a_\iota) < \varepsilon$ for every $\beta < \iota < \alpha$; $(a_\iota)_{\iota<\alpha}$ is *continuous* if $\lim_{\iota\to\lambda} a_\iota = a_\lambda$ for each limit ordinal $\lambda < \alpha$. The sequence $(a_\iota)_{\iota<\alpha}$ is called *Cauchy* if, for any $\varepsilon \in \mathbb{R}^+$, there is a $\beta < \alpha$ such that, for all $\beta < \iota < \gamma < \alpha$, we have that $\mathbf{d}(a_\iota, a_\gamma) < \varepsilon$. A metric space is called *complete* if each of its non-empty Cauchy sequences converges.

Given two metric spaces $(M_1, \mathbf{d}_1)$ and $(M_2, \mathbf{d}_2)$, a function $\phi\colon M_1 \to M_2$ is called an *isometric embedding* of $(M_1, \mathbf{d}_1)$ into $(M_2, \mathbf{d}_2)$ if it preserves distances, i.e.

$$\mathbf{d}_2(\phi(x), \phi(y)) = \mathbf{d}_1(x,y) \qquad \text{for all } x, y \in M_1.$$

If, additionally, ϕ is bijective, then it is called an *isometry* and the metric spaces $(M_1, \mathbf{d}_1)$ and $(M_2, \mathbf{d}_2)$ are said to be *isometric*.

2.3. *Partial orders*

A *partial order* $\leqslant$ on a set A is a binary relation on A that is *transitive*, *reflexive* and *antisymmetric*. The pair $(A, \leqslant)$ is then called a *partially ordered set*. A subset D of the underlying set A is called *directed* if it is non-empty and each pair of elements in D has an upper bound in D. A partially ordered set $(A, \leqslant)$ is called a *complete partial order* (*cpo*) if it has a least element and each directed set D has a *least upper bound* (*lub*) $\bigsqcup D$. A cpo $(A, \leqslant)$ is called a *complete semilattice* if every *non-empty* set B has *greatest lower bound* (*glb*) $\bigsqcap B$. In particular, this means that for any sequence $(a_\iota)_{\iota<\alpha}$ in a complete semilattice, its *limit inferior*, defined by $\liminf_{\iota\to\alpha} a_\iota = \bigsqcup_{\beta<\alpha}\left(\bigsqcap_{\beta\leqslant\iota<\alpha} a_\iota\right)$, exists.

There is also a different characterisation of complete semilattices in terms of bounded complete cpos: a partially ordered set $(A, \leqslant)$ is called *bounded complete* if each set $B \subseteq A$ that has an upper bound in A also has a lub in A.

Proposition 2.1 (Complete semilattice, Kahn and Plotkin (1993)). Given a cpo $(A, \leqslant)$, the following are equivalent:

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

(i) $(A, \leqslant)$ is a complete semilattice.
(ii) $(A, \leqslant)$ is bounded complete.

Given two partially ordered sets $(A, \leqslant_A)$ and $(B, \leqslant_B)$, a function $\phi\colon A \to B$ is called *monotonic* if $a_1 \leqslant_A a_2$ implies $\phi(a_1) \leqslant_B \phi(a_2)$. In particular, a sequence $(b_\iota)_{\iota<\alpha}$ in $(B, \leqslant_B)$ is called monotonic if $\iota \leqslant \gamma < \alpha$ implies $b_\iota \leqslant_B b_\gamma$. An *order isomorphism* from $(A, \leqslant_A)$ to $(B, \leqslant_B)$ is a monotonic function $\phi\colon A \to B$ such that there is a monotonic function $\psi\colon B \to A$ which is the inverse of ϕ, i.e. $\psi \circ \phi$ and $\phi \circ \psi$ are identity functions on A and B, respectively. If there is an order isomorphism from $(A, \leqslant_A)$ to $(B, \leqslant_B)$, then $(A, \leqslant_A)$ and $(B, \leqslant_B)$ are called *order isomorphic*.

With the prefix order $\leqslant$ on sequences, we can generalise concatenation to arbitrary sequences of sequences: Let $(S_\iota)_{\iota<\alpha}$ be a sequence of sequences in some set A. The concatenation of $(S_\iota)_{\iota<\alpha}$, written $\prod_{\iota<\alpha} S_\iota$, is recursively defined as the empty sequence $\langle\rangle$ if $\alpha = 0$, $\left(\prod_{\iota<\alpha'} S_\iota\right) \cdot S_{\alpha'}$ if $\alpha = \alpha' + 1$, and $\bigsqcup_{\gamma<\alpha} \prod_{\iota<\gamma} S_\iota$ if α is a limit ordinal.

2.4. *Terms*

Since we are interested in the infinitary calculus of term rewriting, we consider the set $\mathcal{T}^\infty(\Sigma)$ of (potentially infinite) *terms* over some *signature* Σ. A *signature* Σ is a countable set of symbols. Each symbol f has an associated arity $\mathsf{ar}(f) \in \mathbb{N}$, and we write $\Sigma^{(n)}$ for the set of symbols in Σ which have arity n. The set $\mathcal{T}^\infty(\Sigma)$ is defined as the *greatest* set such that $t \in \mathcal{T}^\infty(\Sigma)$ implies $t = f(t_0, \ldots, t_{k-1})$ for some $f \in \Sigma^{(k)}$ and $t_0, \ldots, t_{k-1} \in \mathcal{T}^\infty(\Sigma)$. For each nullary symbol $c \in \Sigma^{(0)}$, we write c for the term $c()$. For a term $t \in \mathcal{T}^\infty(\Sigma)$, we use the notation $\mathcal{P}(t)$ to denote the *set of positions* in t. $\mathcal{P}(t)$ is the least subset of $\mathbb{N}^*$ such that $\langle\rangle \in \mathcal{P}(t)$ and $\langle i\rangle \cdot \pi \in \mathcal{P}(t)$ if $t = f(t_0, \ldots, t_{k-1})$ with $0 \leqslant i < k$ and $\pi \in \mathcal{P}(t_i)$. For terms $s, t \in \mathcal{T}^\infty(\Sigma)$ and a position $\pi \in \mathcal{P}(t)$, we write $t|_\pi$ for the *subterm* of t at π, $t(\pi)$ for the function symbol in t at π, and $t[s]_\pi$ for the term t with the subterm at π replaced by s. The set $\mathcal{T}(\Sigma)$ of *finite terms* is the set of terms $t \in \mathcal{T}^\infty(\Sigma)$ for which $\mathcal{P}(t)$ is a finite set.

On $\mathcal{T}^\infty(\Sigma)$, a similarity measure $\mathsf{sim}\colon \mathcal{T}^\infty(\Sigma) \times \mathcal{T}^\infty(\Sigma) \to \omega + 1$ can be defined by setting

$$\mathsf{sim}(s,t) = \min\left\{|\pi| \mid \pi \in \mathcal{P}(s) \cap \mathcal{P}(t), s(\pi) \neq t(\pi)\right\} \cup \{\omega\} \qquad \text{for } s, t \in \mathcal{T}^\infty(\Sigma).$$

That is, $\mathsf{sim}(s,t)$ is the minimal depth at which s and t differ, or ω if $s = t$. Based on this, a distance function $\mathbf{d}$ can be defined by $\mathbf{d}(s,t) = 2^{-\mathsf{sim}(s,t)}$, where we interpret $2^{-\omega}$ as 0. The pair $(\mathcal{T}^\infty(\Sigma), \mathbf{d})$ is known to form a complete ultrametric space (Arnold and Nivat 1980). *Partial terms*, i.e. terms over signature $\Sigma_\perp = \Sigma \uplus \{\perp\}$ with $\perp$ a fresh nullary symbol, can be endowed with a binary relation $\leqslant_\perp$ by defining $s \leqslant_\perp t$ iff s can be obtained from t by replacing some subterm occurrences in t by $\perp$. Interpreting the term $\perp$ as denoting 'undefined,' $\leqslant_\perp$ can be read as 'is less defined than.' The pair $(\mathcal{T}^\infty(\Sigma_\perp), \leqslant_\perp)$ is known to form a complete semilattice (Goguen et al. 1977). When dealing with terms in $\mathcal{T}^\infty(\Sigma_\perp)$, we call terms that do not contain the symbol $\perp$, i.e. terms that are contained in $\mathcal{T}^\infty(\Sigma)$, *total*.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

2.5. *Term rewriting systems*

For term rewriting systems we have to consider terms with variables. To this end, we assume a countably infinite set $\mathcal{V}$ of variable symbols and extend a signature Σ to a signature $\Sigma_{\mathcal{V}} = \Sigma \uplus \mathcal{V}$ with variable symbols in $\mathcal{V}$ as nullary symbols. Instead of $\mathcal{T}^{\infty}(\Sigma_{\mathcal{V}})$, we also write $\mathcal{T}^{\infty}(\Sigma, \mathcal{V})$. A term rewriting system (TRS) $\mathcal{R}$ is a pair (Σ, R) consisting of a signature Σ and a set R of *term rewrite rules* of the form $l \to r$ with $l \in \mathcal{T}^{\infty}(\Sigma, \mathcal{V}) \setminus \mathcal{V}$ and $r \in \mathcal{T}^{\infty}(\Sigma, \mathcal{V})$ such that all variables occurring in r also occur in l. Note that both the left- and the right-hand side may be infinite. We usually use x, y, z and primed or indexed variants thereof to denote variables in $\mathcal{V}$.

Similar to the setting of finitary term rewriting, every TRS $\mathcal{R}$ defines a rewrite relation $\to_{\mathcal{R}}$ on terms in $\mathcal{T}^{\infty}(\Sigma)$ as follows:

$$s \to_{\mathcal{R}} t \iff \exists \pi \in \mathcal{P}(s), l \to r \in R, \text{substitution } \sigma : \ s|_{\pi} = l\sigma, t = s[r\sigma]_{\pi}.$$

Instead of $s \to_{\mathcal{R}} t$, we sometimes write $s \to_{\pi,\rho} t$ in order to indicate the applied rule ρ and the position π, or simply $s \to t$. The subterm $s|_{\pi}$ is called a ρ-*redex* or simply *redex*, $r\sigma$ its *contractum*, and $s|_{\pi}$ is said to be *contracted* to $r\sigma$.

2.6. *Convergence of transfinite term reductions*

At first, we look at the metric-based approach of infinitary term rewriting (Dershowitz et al. 1991; Kennaway and de Vries 2003). The convergence of an infinite reduction is determined by the convergence of the underlying sequence of terms in the metric space $(\mathcal{T}^{\infty}(\Sigma), \mathbf{d})$.

A *reduction* in a TRS $\mathcal{R}$, is a sequence $S = (t_{\iota} \to_{\mathcal{R}} t_{\iota+1})_{\iota<\alpha}$ of rewriting steps in $\mathcal{R}$. The sequence $(t_{\iota})_{\iota<\widehat{\alpha}}$ is the underlying sequence of terms, where $\widehat{\alpha} = \alpha$ if α is a limit ordinal, and $\widehat{\alpha} = \alpha + 1$ otherwise. The reduction S is called *weakly m-continuous*, written $S : t_0 \xrightarrow{m} \dots$, if the underlying sequence of terms $(t_{\iota})_{\iota<\widehat{\alpha}}$, is continuous, i.e. $\lim_{\iota\to\lambda} t_{\iota} = t_{\lambda}$ for each limit ordinal $\lambda < \alpha$. The reduction S is said to *weakly m-converge* to a term t, written $S : t_0 \xrightarrow{m} t$, if it is weakly m-continuous and the underlying sequence of terms converges to t, i.e. $\lim_{\iota\to\widehat{\alpha}} t_{\iota} = t$.

Example 2.1. Consider the TRS $\mathcal{R}$ containing the rule $\rho : \ x :: y :: z \to y :: x :: y :: z$, where $::$ is a binary symbol that we write infix and assume to associate to the right. That is, in its explicitly parenthesised form ρ reads $x :: (y :: z) \to y :: (x :: (y :: z))$. Think of the $::$ symbol as the list constructor *cons*. Using the rule ρ, we have the following reduction S of length ω:

$$S : \ a :: a :: c \to a :: a :: \underline{a} :: c \to a :: a :: a :: \underline{a} :: c \to a :: a :: a :: a :: \underline{a} :: c \to \dots$$

The position at which two consecutive terms differ – indicated by the underlining – moves deeper and deeper into the term structure during the reduction S. Hence, the underlying sequence of terms converges to the infinite term s satisfying the equation $s = a :: s$, i.e. $s = a :: a :: a :: \dots$. This means that S weakly m-converges to s.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Now consider the starting term $a :: b :: c$. By repeatedly applying ρ at the root, we obtain the following reduction:

$$T : a :: b :: c \to \underline{b} :: a :: b :: c \to \underline{a} :: b :: a :: b :: c \to \underline{b} :: a :: b :: a :: b :: c \to \dots$$

The difference between consecutive terms remains right at the root position. Hence, the underlying sequence of terms is not Cauchy and, therefore, does not converge. Consequently, T does not weakly m-converge.

However, we can form a weakly m-converging reduction starting from the term $a :: b :: c$ by applying the rule ρ at increasingly deep positions:

$$T' : a :: b :: c \to \underline{b} :: a :: b :: c \to b :: \underline{b} :: a :: b :: c \to b :: b :: \underline{b} :: a :: b :: c \to \dots$$

The reduction T' weakly m-converges to the infinite term $t' = b :: b :: b :: \dots$.

In the partial order approach of infinitary rewriting (Bahr 2010, 2014), convergence is defined in terms of the limit inferior in the partially ordered set $(\mathcal{T}^\infty(\Sigma_\perp), \leqslant_\perp)$: A reduction $S = (t_\iota \to_\mathcal{R} t_{\iota+1})_{\iota<\alpha}$ of *partial terms* is called *weakly p-continuous*, written $S : t_0 \overset{p}{\hookrightarrow} \dots$, if $\liminf_{\iota<\lambda} t_\iota = t_\lambda$ for each limit ordinal $\lambda < \alpha$. The reduction S is said to *weakly p-converge* to a term t, written $S : t_0 \overset{p}{\hookrightarrow} t$, if it is weakly p-continuous and $\liminf_{\iota<\widehat{\alpha}} t_\iota = t$.

The distinguishing feature of the partial order approach is that, due to the complete semilattice structure of $(\mathcal{T}^\infty(\Sigma_\perp), \leqslant_\perp)$, each continuous reduction also converges. Intuitively, weak p-convergence on terms describes an approximation process. To this end, the partial order $\leqslant_\perp$ captures a notion of *information preservation*: $s \leqslant_\perp t$ iff t contains at least the same information as s does but potentially more. A monotonic sequence of terms $t_0 \leqslant_\perp t_1 \leqslant_\perp \dots$ thus approximates the information contained in $t = \bigsqcup_{i<\omega} t_i$: Any finite part of t is contained is some t_i and subsequently remains stable in $t_{i+1}, t_{i+1}, t_{i+2}, \dots$. Given this reading of $\leqslant_\perp$, the glb $\bigsqcap T$ of a set of terms T captures the common (non-contradicting) information of the terms in T. Leveraging this property of the partial order $\leqslant_\perp$, a sequence of terms $(s_i)_{i<\omega}$ that is not necessarily monotonic can be turned into a monotonic sequence $(t_j)_{j<\omega}$ by setting $t_j = \bigsqcap_{j\leqslant i<\omega} s_i$. That is, each t_j contains exactly the information that remains stable in $(s_i)_{i<\omega}$ from j onwards. Hence, the limit inferior $\liminf_{i\to\omega} s_i = \bigsqcup_{j<\omega} \bigsqcap_{j\leqslant i<\omega} s_i$ is the term that contains the accumulated information that eventually remains stable in $(s_i)_{i<\omega}$. This is expressed as an approximation of the monotonically increasing information that remains stable from some point on.

Example 2.2. Reconsider the rule ρ and its induced reduction S from Example 2.1. The reduction S also weakly p-converges to s, i.e. $\liminf_{i\to\omega} s_i$ for $(s_i)_{i<\omega}$ the underlying sequence of terms in S. To see this, consider the sequence $(t_j)_{j<\omega}$ of terms $t_j = \bigsqcap_{j\leqslant i<\omega} s_i$ each of which intuitively encodes the information that remains stable from j onwards:

$$a :: a :: \perp, \quad a :: a :: a :: \perp, \quad a :: a :: a :: a :: \perp, \quad \dots$$

This sequence of terms approximates $s = a :: a :: a :: \dots$ in the sense that $s = \bigsqcup_{j<\omega} t_j$. Likewise, also the reduction T' from Example 2.1 weakly p-converges to the term $t' = b :: b :: b :: \dots$. The sequence of stable information of T' is

$$\perp :: \perp :: \perp, \quad b :: \perp :: \perp :: \perp, \quad b :: b :: \perp :: \perp :: \perp, \quad \dots$$

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

As we have seen, the reduction T from Example 2.1 does not weakly m-converge. However, since T it is trivially weakly p-continuous, it is weakly p-converging. The corresponding sequence of stable information is

$$\bot :: \bot :: \bot, \quad \bot :: \bot :: \bot :: \bot, \quad \bot :: \bot :: \bot :: \bot :: \bot, \quad \dots$$

This sequence approximates the term $t = \bot :: \bot :: \bot :: \dots$, and we thus have that T weakly p-converges to t.

The relation between weak m- and p-convergence illustrated in the examples above is characteristic: Weak p-convergence is a conservative extension of weak m-convergence. In order to qualify this, we say that a reduction $S = (t_\iota \to t_{\iota+1})_{\iota<\alpha}$ weakly p-converges to t in $\mathcal{T}^\infty(\Sigma)$ if S weakly p-converges to t and t as well as each t_ι with $\iota < \widehat{\alpha}$ is in $\mathcal{T}^\infty(\Sigma)$. Analogously, we say that S is weakly p-continuous in $\mathcal{T}^\infty(\Sigma)$ if S is weakly p-continuous and each t_ι with $\iota < \widehat{\alpha}$ is in $\mathcal{T}^\infty(\Sigma)$. We then have the following correspondence between m- and p-convergence:

Theorem 2.1 (p-convergence in $\mathcal{T}^\infty(\Sigma)$ = m-convergence, Bahr (2009)). For every reduction S in a TRS, the following equivalences hold:

(i) $S : s \overset{p}{\hookrightarrow} \dots$ in $\mathcal{T}^\infty(\Sigma)$ iff $S : s \overset{m}{\hookrightarrow} \dots$.
(ii) $S : s \overset{p}{\twoheadrightarrow} t$ in $\mathcal{T}^\infty(\Sigma)$ iff $S : s \overset{m}{\twoheadrightarrow} t$.

Kennaway (1992) and Bahr (2010) investigated abstract models of infinitary rewriting based on metric spaces and partially ordered sets, respectively. We will take these abstract models as a basis to formulate a theory of infinitary term graph reductions. The key question that we have to address is what an appropriate metric space and partial order on term graphs looks like.

3. Graphs and term graphs

This section provides the basic notions for term graphs and more generally for graphs. We shall use the same basic framework of term graphs as in our previous work (Bahr 2012b), where full proofs of the propositions given in this section can be found.

Terms over a signature, say Σ, can be thought of as rooted trees whose nodes are labelled with symbols from Σ. Moreover, in these trees, a node labelled with a k-ary symbol is restricted to have out-degree k and the outgoing edges are ordered. In this way, the ith successor of a node labelled with a symbol f is interpreted as the root node of the subtree that represents the ith argument of f. For example, consider the term $f(a, h(a, b))$. The corresponding representation as a tree is shown in Figure 1a.

In term graphs, the restriction to a tree structure is abolished. The corresponding notion of term graphs we are using is taken from Barendregt et al. (1987).

Definition 3.1 (Graphs). Let Σ be a signature. A *graph* over Σ is a triple $g = (N, \mathsf{lab}, \mathsf{suc})$ consisting of a set N (of *nodes*), a *labelling function* $\mathsf{lab}\colon N \to \Sigma$, and a *successor function* $\mathsf{suc}\colon N \to N^*$ such that $|\mathsf{suc}(n)| = \mathsf{ar}(\mathsf{lab}(n))$ for each node $n \in N$, i.e. a node labelled with a k-ary symbol has precisely k successors. The graph g is called *finite* whenever the

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

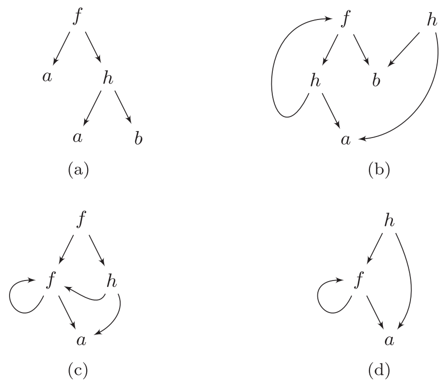


Fig. 1. Tree representation of a term and generalisation to (term) graphs. (a) $f(a, h(a, b))$. (b) A graph. (c) A term graph g. (d) Subterm graph of g.

underlying set N of nodes is finite. If $\mathsf{suc}(n) = \langle n_0, \ldots, n_{k-1} \rangle$, then we write $\mathsf{suc}_i(n)$ for n_i. Moreover, we use the abbreviation $\mathsf{ar}_g(n)$ for the arity $\mathsf{ar}(\mathsf{lab}(n))$ of n.

Example 3.1. Let $\Sigma = \{f/2, h/2, a/0, b/0\}$ be a signature. The graph over Σ, depicted in Figure 1b, is given by the triple $(N, \mathsf{lab}, \mathsf{suc})$ with $N = \{n_0, n_1, n_2, n_3, n_4\}$, $\mathsf{lab}(n_0) = f, \mathsf{lab}(n_1) = \mathsf{lab}(n_4) = h, \mathsf{lab}(n_2) = b, \mathsf{lab}(n_3) = a$ and $\mathsf{suc}(n_0) = \langle n_1, n_2 \rangle, \mathsf{suc}(n_1) = \langle n_0, n_3 \rangle, \mathsf{suc}(n_2) = \mathsf{suc}(n_3) = \langle\rangle, \mathsf{suc}(n_4) = \langle n_2, n_3 \rangle$.

Definition 3.2 (Paths, reachability). Let $g = (N, \mathsf{lab}, \mathsf{suc})$ be a graph and $n, m \in N$.

(i) A *path* in g from n to m is a finite sequence $\pi \in \mathbb{N}^*$ such that either

(a) π is empty and $n = m$, or

(b) $\pi = \langle i \rangle \cdot \pi'$ with $0 \leqslant i < \mathsf{ar}_g(n)$ and the suffix π' a path in g from $\mathsf{suc}_i(n)$ to m.

(ii) If there exists a path in g from n to m, we say that m is *reachable* from n in g.

Definition 3.3 (Term graphs). Given a signature Σ, a *term graph* g over Σ is a tuple $(N, \mathsf{lab}, \mathsf{suc}, r)$ consisting of an *underlying* graph $(N, \mathsf{lab}, \mathsf{suc})$ over Σ whose nodes are all reachable from the *root node* $r \in N$. The term graph g is called *finite* if the underlying graph is finite, i.e. the set N of nodes is finite. The class of all term graphs over Σ is denoted $\mathcal{G}^\infty(\Sigma)$; the class of all finite term graphs over Σ is denoted $\mathcal{G}(\Sigma)$. We use the notation N^g, lab^g, suc^g and r^g to refer to the respective components N,lab, suc and r of g. Given a graph or a term graph h and a node n in h, we write $h|_n$ to denote the sub-term graph of h rooted in n.

Example 3.2. Let $\Sigma = \{f/2, h/2, c/0\}$ be a signature. The term graph over Σ, depicted in Figure 1c, is given by the quadruple $(N, \mathsf{lab}, \mathsf{suc}, r)$, where $N = \{r, n_1, n_2, n_3\}$, $\mathsf{suc}(r) = \langle n_1, n_2 \rangle$, $\mathsf{suc}(n_1) = \mathsf{suc}(n_2) = \langle n_1, n_3 \rangle$, $\mathsf{suc}(n_3) = \langle\rangle$ and $\mathsf{lab}(r) = \mathsf{lab}(n_1) = f$, $\mathsf{lab}(n_2) = h$, $\mathsf{lab}(n_3) = a$. Figure 1d depicts the subterm graph $g|_{n_2}$ of g.

Paths in a graph are not absolute but relative to a starting node. In term graphs, however, we have a distinguished root node from which each node is reachable. Paths

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

relative to the root node correspond to positions in terms and are central for dealing with term graphs:

Definition 3.4 (Positions, depth, cyclicity, trees). Let $g \in \mathcal{G}^\infty(\Sigma)$ and $n \in N^g$.

(i) A *position* of n is a path in the underlying graph of g from r^g to n. The set of all positions in g is denoted $\mathcal{P}(g)$; the set of all positions of n in g is denoted $\mathcal{P}_g(n)$.†

(ii) The *depth* of n in g, denoted $\mathsf{depth}_g(n)$, is the minimum of the lengths of the positions of n in g, i.e. $\mathsf{depth}_g(n) = \min\{|\pi| \mid \pi \in \mathcal{P}_g(n)\}$.

(iii) For a position $\pi \in \mathcal{P}(g)$, we write $\mathsf{node}_g(\pi)$ for the unique node $n \in N^g$ with $\pi \in \mathcal{P}_g(n)$ and $g(\pi)$ for its symbol $\mathsf{lab}^g(n)$.

(iv) A position $\pi \in \mathcal{P}(g)$ is called *cyclic* if there are paths $\pi_1 < \pi_2 \leqslant \pi$ with $\mathsf{node}_g(\pi_1) = \mathsf{node}_g(\pi_2)$. The non-empty path π' with $\pi_1 \cdot \pi' = \pi_2$ is then called a *cycle* of $\mathsf{node}_g(\pi_1)$. A position that is not cyclic is called *acyclic*. The set of all acyclic positions of a node n in g is denoted $\mathcal{P}^a_g(n)$. If g has a cyclic position, g is called cyclic, otherwise g is called acyclic.

(v) The term graph g is called a *term tree* if each node in g has exactly one position.

Note that the labelling function of graphs – and thus term graphs – is *total*. In contrast, Barendregt et al. (1987) considered *open* (term) graphs with a *partial* labelling function such that unlabelled nodes denote holes or variables. This is reflected in their notion of homomorphisms in which the homomorphism condition is suspended for unlabelled nodes.

3.1. *Homomorphisms*

Instead of a partial node labelling function for term graphs, we chose a *syntactic* approach that is closer to the representation in terms: variables, holes and ‘bottoms’ are represented as distinguished syntactic entities. We achieve this on term graphs by making the notion of homomorphisms dependent on a set of constant symbols Δ for which the homomorphism condition is suspended.

Definition 3.5 (Δ-homomorphisms). Let Σ be a signature, $\Delta \subseteq \Sigma^{(0)}$ and $g, h \in \mathcal{G}^\infty(\Sigma)$.

(i) A function $\phi\colon N^g \to N^h$ is called *homomorphic* in $n \in N^g$ if the following holds:

$$\mathsf{lab}^g(n) = \mathsf{lab}^h(\phi(n)) \qquad \text{(labelling)}$$

$$\phi(\mathsf{suc}^g_i(n)) = \mathsf{suc}^h_i(\phi(n)) \quad \text{for all } 0 \leqslant i < \mathsf{ar}_g(n) \qquad \text{(successor)}$$

(ii) A Δ-*homomorphism* ϕ from g to h, denoted $\phi\colon g \to_\Delta h$, is a function $\phi\colon N^g \to N^h$ that is homomorphic in n for all $n \in N^g$ with $\mathsf{lab}^g(n) \notin \Delta$ and satisfies

$$\phi(r^g) = r^h \qquad \text{(root)}$$

† The notion/notation of positions is borrowed from terms: Every position π of a node n corresponds to the subterm represented by n occurring at position π in the unravelling of the term graph to a term.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

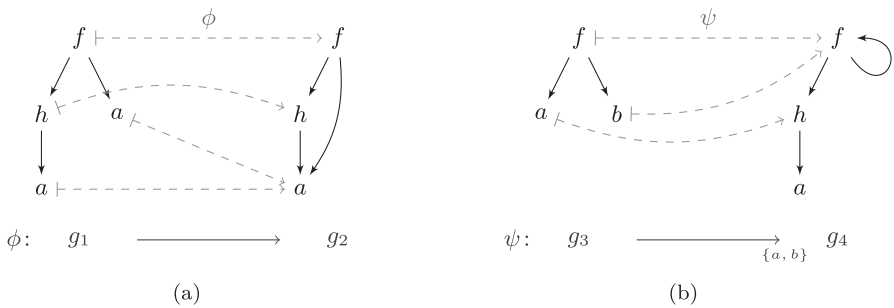


Fig. 2. Δ-homomorphisms. (a) A homomorphism. (b) A $\{a, b\}$-homomorphism.

Note that, for $\Delta = \emptyset$, we get the usual notion of homomorphisms on term graphs (e.g. Barendsen (2003)). The Δ-nodes can be thought of as holes in the term graphs that can be filled with other term graphs. For example, if we have a distinguished set of variable symbols $\mathcal{V} \subseteq \Sigma^{(0)}$, we can use $\mathcal{V}$-homomorphisms to formalise the matching step of term graph rewriting, which requires the instantiation of variables.

Example 3.3. Figure 2 depicts two functions ϕ and ψ. Whereas ϕ is a homomorphism, the function ψ is not a homomorphism since, for example, the node labelled a in g_3 is mapped to a node labelled h in g_3. Nevertheless, ψ is a $\{a, b\}$-homomorphism. Note that Δ-homomorphisms may introduce additional sharing in the target term graph by mapping several nodes in the source to the same node in the target.

Proposition 3.1 (Δ-homomorphism preorder). The Δ-homomorphisms on $\mathcal{G}^\infty(\Sigma)$ form a category that is a preorder, i.e. there is at most one Δ-homomorphism from one term graph to another.

Since Δ-homomorphisms between two given term graphs are unique, we have in particular that any Δ-homomorphism from a term graph g to the same term graph g must be the identity Δ-homomorphism on g. Consequently, whenever there are two Δ-homomorphisms $\phi\colon g \to_\Delta h$ and $\psi\colon h \to_\Delta g$, they are inverses of each other, i.e. Δ-*isomorphisms*. If two term graphs are Δ-*isomorphic*, we write $g \cong_\Delta h$.

For the two special cases $\Delta = \emptyset$ and $\Delta = \{\sigma\}$, we write $\phi\colon g \to h$ and $\phi\colon g \to_\sigma h$, respectively, instead of $\phi\colon g \to_\Delta h$, and we call ϕ a *homomorphism* and a σ-*homomorphism*, respectively. The same convention applies to Δ-isomorphisms.

The structure of positions permits a convenient characterisation of Δ-homomorphisms:

Lemma 3.1 (Characterisation of Δ-homomorphisms). Given $g, h \in \mathcal{G}^\infty(\Sigma)$, a function $\phi\colon N^g \to N^h$ is called a Δ-*homomorphism* $\phi\colon g \to_\Delta h$ iff the following holds for all $n \in N^g$:

(a) $\mathcal{P}_g(n) \subseteq \mathcal{P}_h(\phi(n))$, and
(b) $\mathsf{lab}^g(n) \notin \Delta \implies \mathsf{lab}^g(n) = \mathsf{lab}^h(\phi(n))$.

By Proposition 3.1, there is at most one Δ-homomorphism between two term graphs. The lemma above uniquely defines this Δ-homomorphism: If there is a Δ-homomorphism from g to h, it is defined by $\phi(n) = n'$, where n' is the unique node $n' \in N^h$ with

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

$\mathcal{P}_g(n) \sqsubseteq \mathcal{P}_h(n')$. Moreover, while it is not true for arbitrary Δ-homomorphisms, we have that homomorphisms are surjective.

3.2. *Isomorphisms and isomorphism classes*

When dealing with term graphs, in particular, when studying term graph transformations, we do not want to distinguish between isomorphic term graphs. Distinct but isomorphic term graphs only differ in the naming of nodes and are thus an unwanted artefact of the definition of term graphs. In this way, equality up to isomorphism is similar to α-equivalence of λ-terms and has to be dealt with.

In this section, we characterise isomorphisms and more generally Δ-isomorphisms. From this, we derive two canonical representations of isomorphism classes of term graphs. One is simply a subclass of the class of term graphs, while the other one is based on the structure provided by the positions of term graphs. The relevance of the former representation is derived from the fact that we still have term graphs that can be easily manipulated, whereas the latter is more technical and will be helpful for constructing term graphs up to isomorphism.

From the characterisation of Δ-homomorphisms in Lemma 3.1, we immediately obtain a characterisation of Δ-isomorphisms as follows:

Lemma 3.2 (Characterisation of Δ-isomorphisms). For all $g, h \in \mathcal{G}^\infty(\Sigma)$, a function $\phi\colon N^g \to N^h$ is a Δ-isomorphism iff for all $n \in N^g$

(a) $\mathcal{P}_h(\phi(n)) = \mathcal{P}_g(n)$, and
(b) $\mathsf{lab}^g(n) = \mathsf{lab}^h(\phi(n))$ or $\mathsf{lab}^g(n), \mathsf{lab}^h(\phi(n)) \in \Delta$.

Proof. Immediate consequence of Lemma 3.1 and Proposition 3.1. □

Note that whenever Δ is a singleton set, the condition $\mathsf{lab}^g(n), \mathsf{lab}^h(\phi(n)) \in \Delta$ in the above lemma implies $\mathsf{lab}^g(n) = \mathsf{lab}^h(\phi(n))$. Therefore, we obtain the following corollary:

Corollary 3.1 (σ-isomorphism = isomorphism). Given $g, h \in \mathcal{G}^\infty(\Sigma)$ and $\sigma \in \Sigma^{(0)}$, we have $g \cong h$ iff $g \cong_\sigma h$.

The above equivalence does not hold for Δ-homomorphisms with more than one symbol in Δ: Consider the term graphs $g = a$ and $h = b$ consisting of a single node labelled a b, respectively. While g and h are Δ-isomorphic for $\Delta = \{a, b\}$, they are not isomorphic.

3.2.1. *Canonical term graphs.*

From Lemma 3.2, we learned that isomorphisms between term graphs are mappings that preserve and reflect the positions as well as the labelling of each node. These findings motivate the following definition of canonical term graphs as candidates for representatives of isomorphism classes:

Definition 3.6 (Canonical term graphs). A term graph g is called *canonical* if $n = \mathcal{P}_g(n)$ holds for each $n \in N^g$. That is, each node is the set of its positions in the term graph. The set of all (finite) canonical term graphs over Σ is denoted $\mathcal{G}^\infty_\mathcal{C}(\Sigma)$ (respectively $\mathcal{G}_\mathcal{C}(\Sigma)$). Given a term graph $h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$, its *canonical representative* $\mathcal{C}(h)$ is the canonical term

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

graph given by

$$N^{\mathcal{C}(h)} = \left\{ \mathcal{P}_h(n) \,\middle|\, n \in N^h \right\} \quad r^{\mathcal{C}(h)} = \mathcal{P}_h(r^h) \quad \mathsf{lab}^{\mathcal{C}(h)}(\mathcal{P}_h(n)) = \mathsf{lab}^h(n) \text{ for all } n \in N^h$$
$$\mathsf{suc}_i^{\mathcal{C}(h)}(\mathcal{P}_h(n)) = \mathcal{P}_h(\mathsf{suc}_i^h(n)) \quad \text{for all } n \in N^h, 0 \leqslant i < \mathsf{ar}_h(n).$$

The above definition follows a well-known approach to obtain, for each term graph g, a canonical representative $\mathcal{C}(g)$ (Plump 1999). One can easily see that $\mathcal{C}(g)$ is a well-defined canonical term graph. With this definition, we indeed capture a notion of canonical representatives of isomorphism classes:

Proposition 3.2 (Canonical term graphs). Given $g \in \mathcal{G}^\infty(\Sigma)$, the term graph $\mathcal{C}(g)$ canonically represents the equivalence class $[g]_\cong$. More precisely, it holds that

(i) $g \cong \mathcal{C}(g)$, and
(ii) $g \cong h \quad \text{iff} \quad \mathcal{C}(g) = \mathcal{C}(h)$.

In particular, we have, for all canonical term graphs g, h, that $g = h$ iff $g \cong h$.

Proof. Straightforward consequence of Lemma 3.2. □

3.2.2. *Labelled quotient trees.* Intuitively, term graphs can be thought of as 'terms with sharing,' i.e. terms in which occurrences of the same subterm may be identified. The representation of isomorphic term graphs as *labelled quotient trees*, which we shall study in this section, makes use of this intuition and formalises it. To this end, we introduce an equivalence relation on the positions of a term graph that captures the sharing in a term graph:

Definition 3.7 (Aliasing positions). Given a term graph g and two positions $\pi_1, \pi_2 \in \mathcal{P}(g)$, we say that π_1 and π_2 *alias each other* in g, denoted $\pi_1 \sim_g \pi_2$, if $\mathsf{node}_g(\pi_1) = \mathsf{node}_g(\pi_2)$.

One can easily see that the thus defined relation $\sim_g$ on $\mathcal{P}(g)$ is an equivalence relation. Moreover, the partition on $\mathcal{P}(g)$ induced by $\sim_g$ is simply the set $\{\mathcal{P}_g(n) \,|\, n \in N^g\}$ that contains the sets of positions of nodes in g.

Example 3.4. For the term graph g_2 illustrated in Figure 2a, we have that $\langle 0,0 \rangle \sim_{g_2} \langle 1 \rangle$ as both $\langle 0,0 \rangle$ and $\langle 1 \rangle$ are positions of the a-node in g_2. For the term graph g_4, in Figure 2b, $\langle\rangle \sim_{g_4} \langle 1 \rangle \sim_{g_4} \langle 1,1 \rangle \sim_{g_4} \dots$ since all finite sequences over 1 are positions of the f-node in g_4.

The characterisation of Δ-homomorphisms of Lemma 3.1 can be recast in terms of aliasing positions, which then yields the following characterisation of the *existence* of Δ-homomorphisms:

Lemma 3.3 (Characterisation of Δ-homomorphisms). Given $g, h \in \mathcal{G}^\infty(\Sigma)$, there is a Δ-homomorphism $\phi\colon g \to_\Delta h$ iff, for all $\pi, \pi' \in \mathcal{P}(g)$, we have

(a) $\pi \sim_g \pi' \quad \Longrightarrow \quad \pi \sim_h \pi'$, and
(b) $g(\pi) \notin \Delta \quad \Longrightarrow \quad g(\pi) = h(\pi)$.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Intuitively, Clause (a) states that h has at least as much sharing of nodes as g has, whereas Clause (b) states that h has at least the same non-Δ-labelling as g. In this sense, the above characterisation confirms the intuition about Δ-homomorphisms that we mentioned in Example 3.3, viz. Δ-homomorphisms may only introduce sharing and relabel Δ-nodes. This can be observed in the two Δ-homomorphisms illustrated in Figure 2.

From the above characterisations of the existence of Δ-homomorphisms, we can easily derive the following characterisation of Δ-isomorphisms using the uniqueness of Δ-homomorphisms between two term graphs:

Lemma 3.4 (Characterisation of Δ-isomorphisms). For all $g, h \in \mathcal{G}^\infty(\Sigma)$, we have that $g \cong_\Delta h$ iff

(a) $\sim_g = \sim_h$, and
(b) $g(\pi) = h(\pi)$ or $g(\pi), h(\pi) \in \Delta$ for all $\pi \in \mathcal{P}(g)$.

Lemma 3.4 shows that term graphs can be characterised up to isomorphism by only giving the equivalence $\sim_g$ and the labelling $g(\cdot)\colon \pi \mapsto g(\pi)$ of the involved term graphs. This observation gives rise to the following definition:

Definition 3.8 (Labelled quotient trees). A *labelled quotient tree* over signature Σ is a triple $(P, l, \sim)$ consisting of a non-empty set $P \subseteq \mathbb{N}^*$, a function $l\colon P \to \Sigma$, and an equivalence relation $\sim$ on P that satisfies the following conditions for all $\pi, \pi' \in \mathbb{N}^*$ and $i \in \mathbb{N}$:

$$\pi \cdot \langle i \rangle \in P \implies \pi \in P \quad \text{and} \quad i < \mathsf{ar}(l(\pi)) \qquad \text{(reachability)}$$

$$\pi \sim \pi' \implies \begin{cases} l(\pi) = l(\pi') & \text{and} \\ \pi \cdot \langle i \rangle \sim \pi' \cdot \langle i \rangle & \text{for all } i < \mathsf{ar}(l(\pi)) \end{cases} \qquad \text{(congruence)}$$

In other words, a labelled quotient tree $(P, l, \sim)$ is a ranked tree domain P together with a congruence $\sim$ on it and a labelling function $l\colon P/_\sim \to \Sigma$ that honours the rank. Also, note that since P must be non-empty, the reachability condition implies that $\langle\rangle \in P$.

Example 3.5. The term graph g_2 depicted in Figure 2a is represented up to isomorphism by the labelled quotient tree $(P, l, \sim)$ with $P = \{\langle\rangle, \langle 0\rangle, \langle 0,0\rangle, \langle 1\rangle\}$, $l(\langle\rangle) = f$, $l(\langle 0\rangle) = h$, $l(\langle 0,0\rangle) = l(\langle 1\rangle) = a$ and $\sim$ the least equivalence relation on P with $\langle 0,0\rangle \sim \langle 1\rangle$.

The following lemma confirms that labelled quotient trees uniquely characterise any term graph up to isomorphism:

Lemma 3.5 (Labelled quotient trees are canonical). Each term graph $g \in \mathcal{G}^\infty(\Sigma)$ induces a *canonical labelled quotient tree* $(\mathcal{P}(g), g(\cdot), \sim_g)$ over Σ. Vice versa, for each labelled quotient tree $(P, l, \sim)$ over Σ there is a unique canonical term graph $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ whose canonical labelled quotient tree is $(P, l, \sim)$, i.e. $\mathcal{P}(g) = P$, $g(\pi) = l(\pi)$ for all $\pi \in P$, and $\sim_g = \sim$.

Labelled quotient trees provide a valuable tool for constructing canonical term graphs as we shall see. Nevertheless, the original graph representation remains convenient for practical purposes as it allows a straightforward formalisation of term graph rewriting

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

and provides a finite representation of finite cyclic term graphs, which induce an infinite labelled quotient tree.

3.2.3. *Terms, term trees, and unravelling.* Before we continue, it is instructive to make the correspondence between terms and term graphs clear. First, note that, for each term tree t, the equivalence $\sim_t$ is the identity relation $\mathcal{I}_{\mathcal{P}(t)}$ on $\mathcal{P}(t)$, i.e. $\pi_1 \sim_t \pi_2$ iff $\pi_1 = \pi_2$. Consequently, we have the following one-to-one correspondence between canonical term *trees* and terms: Each term $t \in \mathcal{T}^\infty(\Sigma)$ induces the canonical term tree given by the labelled quotient tree $(\mathcal{P}(t), t(\cdot), \mathcal{I}_{\mathcal{P}(t)})$. For example, the term tree depicted in Figure 1 corresponds to the term $f(a, h(a, b))$. We thus consider the set of terms $\mathcal{T}^\infty(\Sigma)$ as the subset of canonical term trees of $\mathcal{G}^\infty_\mathcal{C}(\Sigma)$.

With this correspondence in mind, we can define the *unravelling* of a term graph g as the unique term t such that there is a homomorphism $\phi\colon\ t \to g$. The unravelling of cyclic term graphs yields infinite terms, e.g. in Figure 6 on page 1399, the term graphs h and h' both unravel to the infinite term $b :: b :: \ldots$. We use the notation $\mathcal{U}(g)$ for the unravelling of g.

4. A simple partial order on term graphs

In this section, we want to establish a partial order suitable for formalising convergence of sequences of canonical term graphs similarly to weak p-convergence on terms.

Recall that weak p-convergence on TRSs is based on a partial order $\leqslant_\perp$ on the set $\mathcal{T}^\infty(\Sigma_\perp)$ of *partial* terms. The partial order $\leqslant_\perp$ instantiates occurrences of $\perp$ from left to right, i.e. $s \leqslant_\perp t$ iff t is obtained by replacing occurrences of $\perp$ in s by arbitrary terms in $\mathcal{T}^\infty(\Sigma_\perp)$.

Analogously, we will consider the class of *partial term graphs* simply as term graphs over the signature $\Sigma_\perp = \Sigma \uplus \{\perp\}$. In order to generalise the partial order $\leqslant_\perp$ to term graphs, we need to formalise the instantiation of occurrences of $\perp$ in term graphs. To this end, we will look more closely at Δ-homomorphisms with $\Delta = \{\perp\}$, or $\perp$-homomorphisms for short. A $\perp$-homomorphism $\phi\colon g \to_\perp h$ maps each node in g to a node in h while 'preserving its structure.' Except for nodes labelled $\perp$ this also includes preserving the labelling. This exception to the homomorphism condition allows the $\perp$-homomorphism ϕ to instantiate each $\perp$-node in g with an arbitrary node in h.

Therefore, we shall use $\perp$-homomorphisms as the basis for generalising $\leqslant_\perp$ to canonical partial term graphs. This approach is based on the observation that $\perp$-homomorphisms characterise the partial order $\leqslant_\perp$ on terms. Considering terms as canonical term trees, we obtain the following characterisation of $\leqslant_\perp$ on terms $s, t \in \mathcal{T}^\infty(\Sigma_\perp)$:

$$s \leqslant_\perp t \iff \text{there is a } \perp\text{-homomorphism } \phi\colon\ s \to_\perp t.$$

Embodying a natural concept on term graphs, $\perp$-homomorphisms thus constitute the ideal tool to define a partial order on canonical partial term graphs that generalises $\leqslant_\perp$.

In this paper, we focus on the simplest among these partial orders on term graphs:

Definition 4.1 (Simple partial order $\leqslant^\mathsf{S}_\perp$). The relation $\leqslant^\mathsf{S}_\perp$ on $\mathcal{G}^\infty(\Sigma_\perp)$ is defined as follows: $g \leqslant^\mathsf{S}_\perp h$ iff there is a $\perp$-homomorphism $\phi\colon\ g \to_\perp h$.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

One of our objective is to argue that the simple partial order $\leqslant^{\mathsf{S}}_{\perp}$ is indeed a suitable structure for deriving a notion of convergence on term graphs in general and for infinitary term graph rewriting in particular.

Due to the preorder structure of $\perp$-homomorphisms on term graphs and the characterisation of isomorphisms as given by Corollary 3.1, the relation $\leqslant^{\mathsf{S}}_{\perp}$ forms a partial order if restricted to canonical term graphs.

Proposition 4.1 (Simple partial order $\leqslant^{\mathsf{S}}_{\perp}$). The relation $\leqslant^{\mathsf{S}}_{\perp}$ is a partial order on $\mathcal{G}^{\infty}_{\mathcal{C}}(\Sigma_{\perp})$.

Proof. Transitivity and reflexivity of $\leqslant^{\mathsf{S}}_{\perp}$ follows immediately from Proposition 3.1. For antisymmetry, consider $g, h \in \mathcal{G}^{\infty}_{\mathcal{C}}(\Sigma_{\perp})$ with $g \leqslant^{\mathsf{S}}_{\perp} h$ and $h \leqslant^{\mathsf{S}}_{\perp} g$. Then, by Proposition 3.1, $g \cong_{\perp} h$. This is equivalent to $g \cong h$ by Corollary 3.1 from which we can conclude $g = h$ using Proposition 3.2. □

Before we study the properties of the partial order $\leqslant^{\mathsf{S}}_{\perp}$, it is helpful to make its characterisation in terms of labelled quotient trees explicit:

Corollary 4.1 (Characterisation of $\leqslant^{\mathsf{S}}_{\perp}$). Let $g, h \in \mathcal{G}^{\infty}(\Sigma_{\perp})$. Then, $g \leqslant^{\mathsf{S}}_{\perp} h$ iff the following conditions are met:

(a) $\pi \sim_g \pi' \quad \Longrightarrow \quad \pi \sim_h \pi' \quad$ for all $\pi, \pi' \in \mathcal{P}(g)$.
(b) $g(\pi) = h(\pi) \quad$ for all $\pi \in \mathcal{P}(g)$ with $g(\pi) \in \Sigma$.

Proof. This follows immediately from Lemma 3.3. □

Note that the partial order $\leqslant_{\perp}$ on terms is entirely characterised by (b). In other words, the partial order $\leqslant^{\mathsf{S}}_{\perp}$ is a combination of the partial order $\leqslant_{\perp}$ imposed on the underlying tree structure of term graphs (i.e. their unravelling) and the preservation of sharing as stipulated by (a).

In order to reflect on the merit of the partial order $\leqslant^{\mathsf{S}}_{\perp}$ as a suitable basis for a notion of convergence on term graphs, recall the characteristics of the partial order-based notion of convergence for terms: Weak p-convergence on terms is based on the ability of the partial order $\leqslant_{\perp}$ to capture *information preservation* between terms – $s \leqslant_{\perp} t$ means that t contains at least the same information as s does. The limit inferior – and thus weak p-convergence – comprises the accumulated information that eventually remains stable along a sequence. Following the approach on terms, a partial order suitable as a basis for convergence for term graph rewriting, has to capture an appropriate notion of information preservation as well.

One has to keep in mind, however, that term graphs encode an additional dimension of information through *sharing* of nodes, i.e. nodes with multiple positions. Since $\leqslant^{\mathsf{S}}_{\perp}$ specialises to $\leqslant_{\perp}$ on terms, it does preserve the information on the tree structure in the same way as $\leqslant_{\perp}$ does. The difficult part is to determine the right approach to the role of sharing.

Indeed, $\perp$-homomorphisms instantiate occurrences of $\perp$ and are thereby able to introduce new information. But they also introduce sharing by mapping different nodes to the same target node: For the term graphs g_0 and g_1 in Figure 3, we have an obvious

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

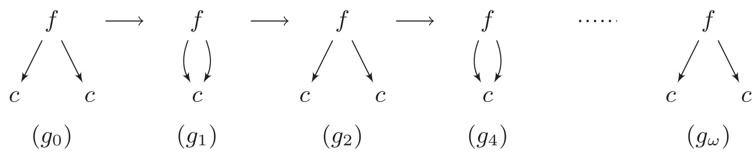


Fig. 3. Limit inferior in the presence of acyclic sharing.

$\bot$-homomorphism – in fact a homomorphism – $\phi\colon g_0 \to_\bot g_1$ and thus $g_0 \leqslant_\bot^{\mathsf{S}} g_1$. However, this homomorphism ϕ maps both c-nodes in g_0 to the single c-node in g_1.

There are at least two different ways to interpret the differences in g_0 and g_1. The first one dismisses $\leqslant_\bot^{\mathsf{S}}$ as a partial order suitable for our purposes: The term graphs g_0 and g_1 contain contradicting information. While in g_0 the two children of the f-node are distinct, they are identical in g_1. We adopted this view in our previous work on convergence for term graphs (Bahr 2012b), where we studied a more rigid partial order $\leqslant_\bot^{\mathsf{R}}$ for which g_0 and g_1 are indeed incomparable (cf. Definition 4.2 below). The second view, which we will adopt in this paper, does not see g_0 and g_1 in contradiction. Both show the f-nodes with two successors, both of which are labelled with c. The term graph g_1 merely contains the additional piece of information that the two successor nodes of the f-node are identical. Hence, $g_0 \leqslant_\bot^{\mathsf{S}} g_1$.

The rest of this section is concerned with showing that the partial order $\leqslant_\bot^{\mathsf{S}}$ has indeed the properties that make it a suitable basis for weak p-convergence, i.e. that it forms a complete semilattice. At first, we show its cpo structure:

Theorem 4.1. The partially ordered set $(\mathcal{G}_\mathcal{C}^\infty(\Sigma_\bot), \leqslant_\bot^{\mathsf{S}})$ is a cpo. In particular, it has the least element $\bot$, and the lub of a directed set G is given by the following labelled quotient tree $(P, l, \sim)$:

$$P = \bigcup_{g \in G} \mathcal{P}(g) \qquad \sim\; = \bigcup_{g \in G} \sim_g \qquad l(\pi) = \begin{cases} f & \text{if } f \in \Sigma \text{ and } \exists g \in G.\ g(\pi) = f \\ \bot & \text{otherwise} \end{cases}$$

Proof. The least element of $\leqslant_\bot^{\mathsf{S}}$ is obviously $\bot$. Hence, it remains to be shown that each directed subset G of $\mathcal{G}_\mathcal{C}^\infty(\Sigma_\bot)$ has a lub $\overline{g}$ given by the labelled quotient tree $(P, l, \sim)$ defined above. To this end, we will make extensive use of Corollary 4.1 using (a) and (b) to refer to its corresponding conditions.

At first, we need to show that l is indeed well-defined. For this purpose, let $g_1, g_2 \in G$ and $\pi \in \mathcal{P}(g_1) \cap \mathcal{P}(g_2)$ with $g_1(\pi), g_2(\pi) \in \Sigma$. Since G is directed, there is some $g \in G$ such that $g_1, g_2 \leqslant_\bot^{\mathsf{S}} g$. By (b), we can conclude $g_1(\pi) = g(\pi) = g_2(\pi)$.

Next, we show that $(P, l, \sim)$ is indeed a labelled quotient tree. Recall that $\sim$ needs to be an equivalence relation. For the reflexivity, assume that $\pi \in P$. Then there is some $g \in G$ with $\pi \in \mathcal{P}(g)$. Since $\sim_g$ is an equivalence relation, $\pi \sim_g \pi$ must hold and, therefore, $\pi \sim \pi$. For the symmetry, assume that $\pi_1 \sim \pi_2$. Then there is some $g \in G$ such that $\pi_1 \sim_g \pi_2$. Hence, we get $\pi_2 \sim_g \pi_1$ and, consequently, $\pi_2 \sim \pi_1$. In order to show transitivity, assume that $\pi_1 \sim \pi_2, \pi_2 \sim \pi_3$. That is, there are $g_1, g_2 \in G$ with $\pi_1 \sim_{g_1} \pi_2$ and $\pi_2 \sim_{g_2} \pi_3$. Since G is directed, we find some $g \in G$ such that $g_1, g_2 \leqslant_\bot^{\mathsf{S}} g$. By (a), this implies that also $\pi_1 \sim_g \pi_2$ and $\pi_2 \sim_g \pi_3$. Hence, $\pi_1 \sim_g \pi_3$ and, therefore, $\pi_1 \sim \pi_3$.

For the reachability condition, let $\pi \cdot \langle i \rangle \in P$. That is, there is a $g \in G$ with $\pi \cdot \langle i \rangle \in \mathcal{P}(g)$. Hence, $\pi \in \mathcal{P}(g)$, which in turn implies $\pi \in P$. Moreover, $\pi \cdot \langle i \rangle \in \mathcal{P}(g)$ implies that $i < \mathsf{ar}(g(\pi))$. Since $g(\pi)$ cannot be a nullary symbol and in particular not $\bot$, we obtain that $l(\pi) = g(\pi)$. Hence, $i < \mathsf{ar}(l(\pi))$.

For the congruence condition, assume that $\pi_1 \sim \pi_2$ and that $l(\pi_1) = f$. If $f \in \Sigma$, then there are $g_1, g_2 \in G$ with $\pi_1 \sim_{g_1} \pi_2$ and $g_2(\pi_1) = f$. Since G is directed, there is some $g \in G$ such that $g_1, g_2 \leqslant^{\mathsf{S}}_{\bot} g$. Hence, by (a) and (b), we have $\pi_1 \sim_g \pi_2$ and $g(\pi_1) = f$. Using Lemma 3.5, we can conclude that $g(\pi_2) = g(\pi_1) = f$ and that $\pi_1 \cdot \langle i \rangle \sim_g \pi_2 \cdot \langle i \rangle$ for all $i < \mathsf{ar}(g(\pi_1))$. Because $g \in G$, it holds that $l(\pi_2) = f$ and that $\pi_1 \cdot \langle i \rangle \sim \pi_2 \cdot \langle i \rangle$ for all $i < \mathsf{ar}(l(\pi_1))$. If $f = \bot$, then also $l(\pi_2) = \bot$, for if $l(\pi_2) = f'$ for some $f' \in \Sigma$, then, by the symmetry of $\sim$ and the above argument (for the case $f \in \Sigma$), we would obtain $f = f'$ and, therefore, a contradiction. Since $\bot$ is a nullary symbol, the remainder of the condition is vacuously satisfied.

This shows that $(P, l, \sim)$ is a labelled quotient tree which, by Lemma 3.5, uniquely defines a canonical term graph. In order to show that the thus obtained term graph $\overline{g}$ is an upper bound for G, we have to show that $g \leqslant^{\mathsf{S}}_{\bot} \overline{g}$ for all $g \in G$ by establishing (a) and (b). This is an immediate consequence of the construction of $\overline{g}$.

In the final part of this proof, we will show that $\overline{g}$ is the lub of G. For this purpose, let $\hat{g}$ be an upper bound of G, i.e. $g \leqslant^{\mathsf{S}}_{\bot} \hat{g}$ for all $g \in G$. We will show that $\overline{g} \leqslant^{\mathsf{S}}_{\bot} \hat{g}$ by establishing (a) and (b). For (a), assume that $\pi_1 \sim \pi_2$. Hence, there is some $g \in G$ with $\pi_1 \sim_g \pi_2$. Since by assumption, $g \leqslant^{\mathsf{S}}_{\bot} \widehat{g}$, we can conclude $\pi_1 \sim_{\widehat{g}} \pi_2$ using (a). For (b), assume $\pi \in P$ and $l(\pi) = f \in \Sigma$. Then there is some $g \in G$ with $g(\pi) = f$. Applying (b) then yields $\widehat{g}(\pi) = f$ since $g \leqslant^{\mathsf{S}}_{\bot} \widehat{g}$. □

The following proposition shows that the partial order $\leqslant^{\mathsf{S}}_{\bot}$ also admits glbs of arbitrary non-empty sets:

Proposition 4.2. In the partially ordered set $(\mathcal{G}^{\infty}_{\mathcal{C}}(\Sigma_{\bot}), \leqslant^{\mathsf{S}}_{\bot})$, every non-empty set has a glb. In particular, the glb of a non-empty set G is given by the following labelled quotient tree $(P, l, \sim)$:

$$P = \left\{ \pi \in \bigcap_{g \in G} \mathcal{P}(g) \,\middle|\, \forall \pi' < \pi \exists f \in \Sigma_{\bot} \forall g \in G : g(\pi') = f \right\}$$

$$l(\pi) = \begin{cases} f & \text{if } \forall g \in G : f = g(\pi) \\ \bot & \text{otherwise} \end{cases} \qquad \sim = \bigcap_{g \in G} \sim_g \cap P \times P$$

Proof. At first, we need to prove that $(P, l, \sim)$ is in fact a well-defined labelled quotient tree. That $\sim$ is an equivalence relation follows straightforwardly from the fact that each $\sim_g$ is an equivalence relation.

Next, we show the reachability and congruence properties from Definition 3.8. In order to show the reachability property, assume some $\pi \cdot \langle i \rangle \in P$. Then, for each $\pi' \leqslant \pi$, there is some $f_{\pi'} \in \Sigma_{\bot}$ such that $g(\pi') = f_{\pi'}$ for all $g \in G$. Hence, $\pi \in P$. Moreover, we have in particular that $i < \mathsf{ar}(f_{\pi}) = \mathsf{ar}(l(\pi))$.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

For the congruence condition, assume that $\pi_1 \sim \pi_2$. Hence, $\pi_1 \sim_g \pi_2$ for all $g \in G$. Consequently, we have for each $g \in G$ that $g(\pi_1) = g(\pi_2)$ and that $\pi_1 \cdot \langle i \rangle \sim_g \pi_2 \cdot \langle i \rangle$ for all $i < \mathsf{ar}(g(\pi_1))$. We distinguish two cases: at first, assume that there are some $g_1, g_2 \in G$ with $g_1(\pi_1) \neq g_2(\pi_1)$. Hence, $l(\pi_2) = \bot$. Since we also have that $g_1(\pi_2) = g_1(\pi_1) \neq g_2(\pi_1) = g_2(\pi_2)$, we can conclude that $l(\pi_2) = \bot = l(\pi_1)$. Since $\mathsf{ar}(\bot) = 0$, we are done for this case. Next, consider the alternative case that there is some $f \in \Sigma_\bot$ such that $g(\pi_1) = f$ for all $g \in G$. Consequently, $l(\pi_1) = f$ and since also $g(\pi_2) = g(\pi_1) = f$ for all $g \in G$, we can conclude that $l(\pi_2) = f = l(\pi_1)$. Moreover, we obtain from the initial assumption for this case, that $\pi_1 \cdot \langle i \rangle, \pi_2 \cdot \langle i \rangle \in P$ for all $i < \mathsf{ar}(f)$ which implies that $\pi_1 \cdot \langle i \rangle \sim \pi_2 \cdot \langle i \rangle$ for all $i < \mathsf{ar}(f) = \mathsf{ar}(l(\pi_1))$.

Next, we show that the term graph $\overline{g}$ defined by $(P, l, \sim)$ is a lower bound of G, i.e. that $\overline{g} \leqslant^{\mathsf{S}}_\bot g$ for all $g \in G$. By Corollary 4.1, it suffices to show $\sim \cap P \times P \subseteq \sim_g$ and $l(\pi) = g(\pi)$ for all $\pi \in P$ with $l(\pi) \in \Sigma$. Both conditions follow immediately from the construction of $\overline{g}$.

Finally, we show that $\overline{g}$ is the greatest lower bound of G. To this end, let $\widehat{g} \in \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$ with $\widehat{g} \leqslant^{\mathsf{S}}_\bot g$ for each $g \in G$. We will show that then $\widehat{g} \leqslant^{\mathsf{S}}_\bot \overline{g}$ using Corollary 4.1. At first, we show that $\mathcal{P}(\widehat{g}) \subseteq P$. Let $\pi \in \mathcal{P}(\widehat{g})$. We know that $\widehat{g}(\pi') \in \Sigma$ for all $\pi' < \pi$. According to Corollary 4.1, using the assumption that $\widehat{g} \leqslant^{\mathsf{S}}_\bot g$ for all $g \in G$, we obtain that $g(\pi') = \widehat{g}(\pi')$ for all $\pi' < \pi$. Consequently, $\pi \in P$. Next, we show part (a) of Corollary 4.1. Let $\pi_1, \pi_2 \in \mathcal{P}(\widehat{g}) \subseteq P$ with $\pi_1 \sim_{\widehat{g}} \pi_2$. Hence, using the assumption that $\widehat{g}$ is a lower bound of G, we have $\pi_1 \sim_g \pi_2$ for all $g \in G$ according to Corollary 4.1. Consequently, $\pi_1 \sim \pi_2$. For part (b) of Corollary 4.1, let $\pi \in \mathcal{P}(\widehat{g}) \subseteq P$ with $\widehat{g}(\pi) = f \in \Sigma$. Using Corollary 4.1, we obtain that $g(\pi) = f$ for all $g \in G$. Hence, $l(\pi) = f$. □

From this, we can immediately derive the complete semilattice structure of $\leqslant^{\mathsf{S}}_\bot$:

Theorem 4.2. The partially ordered set $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot), \leqslant^{\mathsf{S}}_\bot)$ forms a complete semilattice.

Proof. Follows from Theorem 4.1 and Proposition 4.2. □

In particular, this means that the limit inferior is defined for every sequence of term graphs. Moreover, from the constructions given in Theorem 4.1 and Proposition 4.2, we can derive the following direct construction of the limit inferior:

Corollary 4.2. The limit inferior of a sequence $(g_\iota)_{\iota<\alpha}$ in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot), \leqslant^{\mathsf{S}}_\bot)$ is given by the following labelled quotient tree $(P, l, \sim)$:

$$P = \bigcup_{\beta<\alpha} \{\pi \in \mathcal{P}(g_\beta) \mid \forall \pi' < \pi \forall \beta \leqslant \iota < \alpha : g_\iota(\pi') = g_\beta(\pi')\}$$

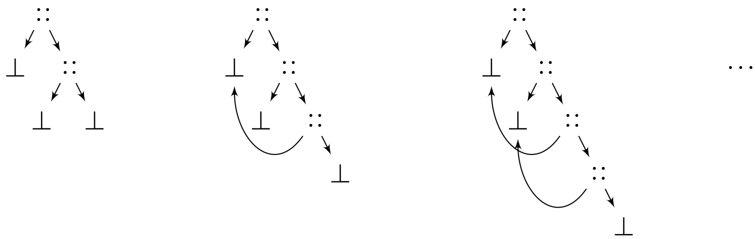
$$\sim = \left(\bigcup_{\beta<\alpha} \bigcap_{\beta\leqslant\iota<\alpha} \sim_{g_\iota} \right) \cap P \times P$$

$$l(\pi) = \begin{cases} g_\beta(\pi) & \text{if } \exists \beta < \alpha \forall \beta \leqslant \iota < \alpha : g_\iota(\pi) = g_\beta(\pi) \\ \bot & \text{otherwise} \end{cases} \qquad \text{for all } \pi \in P.$$

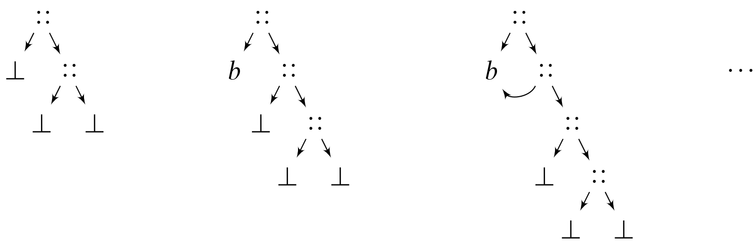
https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

In particular, given $\beta < \alpha$ and $\pi \in \mathcal{P}(g_\beta)$, we have that $g(\pi) = g_\beta(\pi)$ if $g_\iota(\pi') = g_\beta(\pi')$ for all $\pi' \leqslant \pi$ and $\beta \leqslant \iota < \alpha$.

Example 4.1. Figure 6c and d on page 40 illustrate two sequences of term graphs $(g_\iota)_{\iota<\omega}$ and $(h_\iota)_{\iota<\omega}$ together with their limit inferiors g_ω and h_ω, respectively. To see how these limits come about, consider first the sequence of glbs $(\bigsqcap_{\alpha\leqslant\iota<\omega} g_\iota)_{\alpha<\omega}$ of $(g_\iota)_{\iota<\omega}$:



The lub of this sequence of term graphs is the term graph g_ω. The corresponding sequence $(\bigsqcap_{\alpha\leqslant\iota<\omega} h_\iota)_{\alpha<\omega}$ of glbs for $(h_\iota)_{\iota<\omega}$ looks as follows:



With each step, the number of edges into the b-node increases by one and the $\bot$-nodes move further down the graph structure. The lub of this sequence is the term graph h_ω.

Changing acyclic sharing may, however, expose an oddity of the partial order $\leqslant_\bot^{\mathsf{S}}$. Let $(g_\iota)_{\iota<\omega}$ be the sequence of term graphs illustrated in Figure 3. The sequence alternates between g_0 and g_1 which differ only in the sharing of the two arguments of the f function symbol. Hence, there is an obvious homomorphism from g_0 to g_1 and we thus have $g_0 \leqslant_\bot^{\mathsf{S}} g_1$. Therefore, g_0 is the greatest lower bound of every suffix of $(g_\iota)_{\iota<\omega}$, which means that $\liminf_{\iota\to\omega} g_\iota = g_0$.

In our previous work (Bahr 2012b), we have used a partial order $\leqslant_\bot^{\mathsf{R}}$ that is more rigid than $\leqslant_\bot^{\mathsf{S}}$. In the context of this partial order $\leqslant_\bot^{\mathsf{R}}$, limit inferior of the sequence illustrated in Figure 3 changes to the term tree $f(\bot,\bot)$ instead of $f(c,c)$.

Definition 4.2 (Rigid $\bot$-homomorphisms, rigid partial order $\leqslant_\bot^{\mathsf{R}}$).

(i) A $\bot$-homomorphism $\phi\colon g \to_\bot h$ between two term graphs $g, h \in \mathcal{G}^\infty(\Sigma_\bot)$ is called *rigid* if, for all $n \in N^g$ with $\mathsf{lab}^g(n) \in \Sigma$, we have that $\mathcal{P}_g^a(n) = \mathcal{P}_h^a(\phi(n))$, i.e. n and $\phi(n)$ have the same acyclic positions.

(ii) For every $g, h \in \mathcal{G}^\infty(\Sigma_\bot)$, define $g \leqslant_\bot^{\mathsf{R}} h$ iff there is a rigid $\bot$-homomorphism $\phi\colon g \to_\bot h$.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

The difference in the convergence behaviour of $\leqslant_\bot^{\mathsf{S}}$ and $\leqslant_\bot^{\mathsf{R}}$ stems from their difference in dealing with sharing, which we have discussed in the beginning of this section: the partial order $\leqslant_\bot^{\mathsf{S}}$ sees the term graph g_1 as the term graph g_0 with the additional information that the two arguments of f coincide. Since this additional piece of information is not stable throughout the sequence $(g_i)_{i<\omega}$, the limit inferior is only the term graph g_0.

The partial order $\leqslant_\bot^{\mathsf{R}}$, on the other hand, sees the two term graphs g_0 and g_1 in conflict due to the difference in the arguments of f; it is a difference in acyclic sharing. Thus, the sequence $(g_i)_{i<\omega}$ is only stable in the root nodes of the term graphs and the limit inferior is consequently the term tree $f(\bot, \bot)$.

In our previous work (Bahr 2012b), we chose the rigid partial order as there is a metric space that is 'compatible' with it. However, this property of the partial order $\leqslant_\bot^{\mathsf{R}}$ comes at a price: $\leqslant_\bot^{\mathsf{R}}$ is quite restrictive in its ability to represent acyclic sharing. For example, the sequence $(h_\iota)_{\iota<\omega}$ of term graphs depicted in Figure 6d does not have the anticipated limit inferior h_ω but instead the term graph obtained from h_ω by relabelling the b-node with $\bot$.

For the partial order $\leqslant_\bot^{\mathsf{S}}$, we will not be able to find a metric space that is 'compatible' with it in the same way and as a consequence we will not obtain the same correspondence that Theorem 2.1 exposed for infinitary term rewriting. In the following section, we will, however, devise a simple metric space that comes close enough to being 'compatible' with $\leqslant_\bot^{\mathsf{S}}$ such that it is possible to regain the correspondence between p-convergence and m-convergence in the setting of strong convergence (cf. Section 9).

5. A simple metric on term graphs

In this section, we pursue the metric approach to convergence in rewriting systems. To this end, we shall define a metric space on canonical term graphs. We base our approach to defining a metric distance on the definition of the metric distance **d** on terms.

Originally, Arnold and Nivat (1980) used a notion of truncation of terms to define the metric on terms. The truncation of a term t at depth d, denoted $t|d$, replaces all subterms at depth d by $\bot$:

$$t|0 = \bot, \quad f(t_1, \ldots, t_k)|d+1 = f(t_1|d, \ldots, t_k|d), \quad t|\omega = t.$$

For technical reasons, we also define the truncation at depth ω, which does not affect the term at all.

Recall that the metric distance **d** on terms is defined by $\mathbf{d}(s,t) = 2^{-\mathsf{sim}(s,t)}$. The underlying notion of similarity $\mathsf{sim}\colon \mathcal{T}^\infty(\Sigma) \times \mathcal{T}^\infty(\Sigma) \to \omega + 1$ can be characterised via truncations:

$$\mathsf{sim}(s,t) = \max\{d \leqslant \omega \mid s|d = t|d\}$$

We will adopt this approach for term graphs as well. To this end, we will first define abstractly what a truncation on term graphs is and how a metric distance can be derived from it. Then we devise a concrete truncation and show that the induced metric space is in fact complete. We will conclude the section by showing that the metric space we considered is robust in the sense that it is invariant under small changes to the definition of truncation. Last, we contrast this finding with the properties of the complete metric that we have previously studied as a candidate for describing convergence on term graphs (Bahr 2012b).

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

5.1. *Truncation functions*

As we have seen above, the truncation on terms is a function that, depending on a depth value d, transforms a term t to a term $t|d$. We shall generalise this to term graphs and stipulate some axioms that ensure that we can derive a metric distance in the style of Arnold and Nivat (1980):

Definition 5.1 (Truncation function). A family $\tau = (\tau_d : \mathcal{G}^\infty(\Sigma_\perp) \to \mathcal{G}^\infty(\Sigma_\perp))_{d \leqslant \omega}$ of functions on term graphs is called a *truncation function* if it satisfies the following properties for all $g, h \in \mathcal{G}^\infty(\Sigma_\perp)$ and $d \leqslant \omega$:

(a) $\tau_0(g) \cong \perp$,
(b) $\tau_\omega(g) \cong g$, and
(c) $\tau_d(g) \cong \tau_d(h) \implies \tau_e(g) \cong \tau_e(h)$ for all $e < d$.

Note that from axioms (b) and (c) it follows that truncation functions must be defined modulo isomorphism, i.e. $g \cong h$ implies $\tau_d(g) \cong \tau_d(h)$ for all $d \leqslant \omega$.

Given a truncation function, we can define a distance measure in the style of Arnold and Nivat.

Definition 5.2 (Truncation-based similarity/distance). Let τ be a truncation function. The τ-*similarity* is the function $\mathsf{sim}_\tau : \mathcal{G}^\infty(\Sigma_\perp) \times \mathcal{G}^\infty(\Sigma_\perp) \to \omega + 1$ defined by

$$\mathsf{sim}_\tau(g, h) = \max\{d \leqslant \omega \mid \tau_d(g) \cong \tau_d(h)\}.$$

The τ-*distance* is the function $\mathbf{d}_\tau : \mathcal{G}^\infty(\Sigma_\perp) \times \mathcal{G}^\infty(\Sigma_\perp) \to \mathbb{R}_0^+$ defined by $\mathbf{d}_\tau(g,h) = 2^{-\mathsf{sim}_\tau(g,h)}$, where $2^{-\omega}$ is interpreted as 0.

Observe that the similarity $\mathsf{sim}_\tau(g, h)$ induced by a truncation function τ is well-defined since the axiom (a) of Definition 5.1 insures that the set $\{d \leqslant \omega \mid \tau_d(g) \cong \tau_d(h)\}$ is not empty. The following proposition confirms that the τ-distance restricted to $\mathcal{G}_\mathcal{C}^\infty(\Sigma)$ is indeed an ultrametric.

Proposition 5.1 (Truncation-based ultrametric). For each truncation function τ, the τ-distance $\mathbf{d}_\tau$ constitutes an ultrametric on $\mathcal{G}_\mathcal{C}^\infty(\Sigma)$.

Proof. The identity and the symmetry conditions follow by

$$\mathbf{d}_\tau(g,h) = 0 \iff \mathsf{sim}_\tau(g,h) = \omega \iff \tau_\omega(g) \cong \tau_\omega(h) \overset{(*)}{\iff} g \cong h \overset{\text{Prop. 3.2}}{\iff} g = h, \quad \text{and}$$
$$\mathbf{d}_\tau(g,h) = 2^{-\mathsf{sim}_\tau(g,h)} = 2^{-\mathsf{sim}_\tau(h,g)} = \mathbf{d}_\tau(h,g).$$

The equivalence $(*)$ is valid by axiom (b) of Definition 5.1. For the strong triangle condition, we have to show that

$$\mathsf{sim}_\tau(g_1, g_3) \geqslant \min\{\mathsf{sim}_\tau(g_1, g_2), \mathsf{sim}_\tau(g_2, g_3)\}.$$

With $d = \min\{\mathsf{sim}_\tau(g_1, g_2), \mathsf{sim}_\tau(g_2, g_3)\}$, we have, by axiom (c) of Definition 5.1, that $\tau_d(g_1) \cong \tau_d(g_2)$ and $\tau_d(g_2) \cong \tau_d(g_3)$. Since we have that $\tau_d(g_1) \cong \tau_d(g_3)$ then, we can conclude that $\mathsf{sim}_\tau(g_1, g_3) \geqslant d$. □

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Given their particular structure, we can reformulate the characterisation of Cauchy sequences and convergence in metric spaces induced by truncation functions in terms of the truncation function itself:

Lemma 5.1. For each truncation function τ, term graph $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$, and sequence $(g_\iota)_{\iota<\alpha}$ in $\mathcal{G}^\infty_\mathcal{C}(\Sigma)$) the following holds:

(i) $(g_\iota)_{\iota<\alpha}$ is Cauchy in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau)$ iff for each $d < \omega$ there is some $\beta < \alpha$ such that $\tau_d(g_\gamma) \cong \tau_d(g_\iota)$ for all $\beta \leqslant \gamma, \iota < \alpha$.

(ii) $(g_\iota)_{\iota<\alpha}$ converges to g in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau)$ iff for each $d < \omega$ there is some $\beta < \alpha$ such that $\tau_d(g) \cong \tau_d(g_\iota)$ for all $\beta \leqslant \iota < \alpha$.

Proof. We only show (i) as (ii) follows analogously. For 'only if' direction, assume that $(g_\iota)_{\iota<\alpha}$ is Cauchy and that $d < \omega$. We then find some $\beta < \alpha$ such that $\mathbf{d}_\tau(g_\gamma, g_\iota) < 2^{-d}$ for all $\beta \leqslant \gamma, \iota < \alpha$. Hence, we obtain that $\mathsf{sim}_\tau(g_\gamma, g_\iota) > d$ for all $\beta \leqslant \gamma, \iota < \alpha$. That is, $\tau_e(g_\gamma) \cong \tau_e(g_\iota)$ for some $e > d$. According to axiom (c) of Definition 5.1, we can then conclude that $\tau_d(g_\gamma) \cong \tau_d(g_\iota)$ for all $\beta \leqslant \gamma, \iota < \alpha$.

For the 'if' direction, assume some positive real number $\varepsilon \in \mathbb{R}^+$. Then there is some $d < \omega$ with $2^{-d} \leqslant \varepsilon$. By the initial assumption, we find some $\beta < \alpha$ with $\tau_d(g_\gamma) \cong \tau_d(g_\iota)$ for all $\beta \leqslant \gamma, \iota < \alpha$, i.e. $\mathsf{sim}_\tau(g_\gamma, g_\iota) \geqslant d$. Hence, we have that $\mathbf{d}_\tau(g_\gamma, g_\iota) = 2^{-\mathsf{sim}_\tau(g_\gamma, g_\iota)} < 2^{-d} \leqslant \varepsilon$ for all $\beta \leqslant \gamma, \iota < \alpha$. □

5.2. *The simple truncation and its metric space*

In this section, we consider a straightforward truncation function that simply cuts off all nodes at the given depth d. The metric that we obtain from this truncation will be the companion metric for the simple partial order $\leqslant^{\mathsf{S}}_\bot$.

Definition 5.3 (Simple truncation). Let $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and $d \leqslant \omega$. The *simple truncation* $g\dagger d$ of g at d is the term graph defined as follows:

$$N^{g\dagger d} = \{n \in N^g \mid \mathsf{depth}_g(n) \leqslant d\} \qquad r^{g\dagger d} = r^g$$

$$\mathsf{lab}^{g\dagger d}(n) = \begin{cases} \mathsf{lab}^g(n) & \text{if } \mathsf{depth}_g(n) < d \\ \bot & \text{if } \mathsf{depth}_g(n) = d \end{cases} \qquad \mathsf{suc}^{g\dagger d}(n) = \begin{cases} \mathsf{suc}^g(n) & \text{if } \mathsf{depth}_g(n) < d \\ \langle\rangle & \text{if } \mathsf{depth}_g(n) = d \end{cases}$$

One can easily see that the truncated term graph $g\dagger d$ is obtained from g by relabelling all nodes at depth d to $\bot$, removing all their outgoing edges and then removing all nodes that thus become unreachable from the root. This makes the simple truncation a straightforward generalisation of the truncation on terms.

Figure 4 shows a term graph g and its simple truncation at depth $d = 2$. The shaded part of the term graph g comprises the nodes at depth $< d$. Note that a node can get truncated even though some its successor are retained.

The simple truncation indeed induces a truncation function:

Proposition 5.2. Let $\dagger$ be the function with $\dagger_d(g) = g\dagger d$ for all $d \leqslant \omega$. Then, $\dagger$ is a truncation function.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

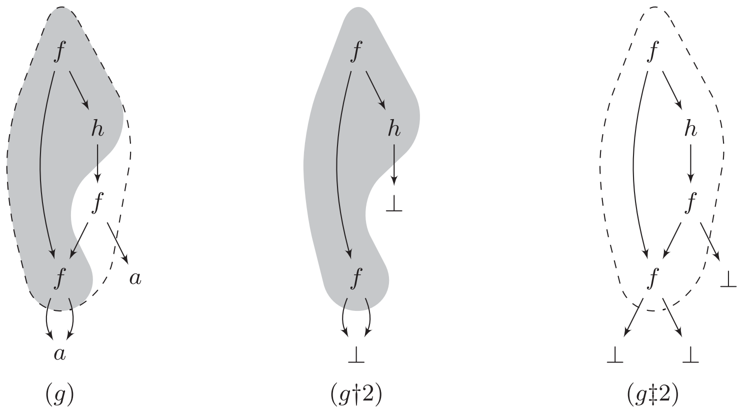


Fig. 4. Comparison of simple and rigid truncation.

Proof. (a) and (b) of Definition 5.1 follow immediately from the construction of the truncation. For (c), assume that $g\dagger d \cong h\dagger d$. Let $0 \leqslant e < d$ and let $\phi\colon g\dagger d \to h\dagger d$ be the witnessing isomorphism. Note that simple truncations preserve the depth of nodes, i.e. $\mathsf{depth}_{g\dagger d}(n) = \mathsf{depth}_g(n)$ for all $n \in N^{g\dagger d}$. This can be shown by a straightforward induction on $\mathsf{depth}_g(n)$. Moreover, by Lemma 3.2 also isomorphisms preserve the depth of nodes. Hence,

$$\mathsf{depth}_h(\phi(n)) = \mathsf{depth}_{h\dagger d}(\phi(n)) = \mathsf{depth}_{g\dagger d}(n) = \mathsf{depth}_g(n) \quad \text{for all } n \in N^{g\dagger d}.$$

Restricting ϕ to the nodes in $g\dagger e$ thus yields an isomorphism from $g\dagger e$ to $h\dagger e$. □

Next, we show that the metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$ that is induced by the truncation function $\dagger$ is in fact complete. To do this, we give a characterisation of the simple truncation in terms of labelled quotient trees.

Lemma 5.2 (Labelled quotient tree of a simple truncation). Let $g \in \mathcal{G}^\infty(\Sigma_\perp)$ and $d \leqslant \omega$. The simple truncation $g\dagger d$ is uniquely determined up to isomorphism by the labelled quotient tree $(P, l, \sim)$ with

(a) $P = \{\pi \in \mathcal{P}(g) \mid \forall \pi_1 < \pi \exists \pi_2 \sim_g \pi_1 \text{ with } |\pi_2| < d\}$,

(b) $l(\pi) = \begin{cases} g(\pi) & \text{if } \exists \pi' \sim_g \pi \text{ with } |\pi'| < d \\ \perp & \text{otherwise,} \end{cases}$

(c) $\sim \,=\, \sim_g \cap\, P \times P$.

Proof. We just have to show that $(P, l, \sim)$ is the labelled quotient tree induced by $g\dagger d$. Then the lemma follows from Lemma 3.5. The case $d = \omega$ is trivial. In the following, we assume that $d < \omega$.

At first, note that

$$\text{for each } \pi \in \mathcal{P}(g\dagger d), \text{ we have that } \pi \in \mathcal{P}(g) \text{ and } \mathsf{node}_{g\dagger d}(\pi) = \mathsf{node}_g(\pi). \qquad (*)$$

This can be shown by an induction on the length of π: the case $\pi = \langle\rangle$ is trivial. If $\pi = \pi' \cdot \langle i \rangle$, let $n = \mathsf{node}_{g\dagger d}(\pi')$ and $m = \mathsf{node}_{g\dagger d}(\pi)$. Hence, $m = \mathsf{suc}_i^{g\dagger d}(n)$ and, by

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

construction of $g\dagger d$, also $m = \mathsf{suc}_i^g(n)$. Since by induction hypothesis $n = \mathsf{node}_g(\pi')$, we can thus conclude that $\pi \in \mathcal{P}(g)$ and that $\mathsf{node}_g(\pi) = m = \mathsf{node}_{g\dagger d}(\pi)$.

(a) $P = \mathcal{P}(g\dagger d)$. For the '$\subseteq$' direction, let $\pi \in P$. We show by induction on the length of π that $\pi \in \mathcal{P}(g\dagger d)$. The case $\pi = \langle\rangle$ is trivial. If $\pi = \pi_1 \cdot \langle i\rangle$, then by induction hypothesis $\pi_1 \in \mathcal{P}(g\dagger d)$. Let $n = \mathsf{node}_{g\dagger d}(\pi_1)$. By (*), we know that $n = \mathsf{node}_g(\pi_1)$. Since $\pi_1 \cdot \langle i\rangle \in P$, there is some $\pi_2 \sim_g \pi_1$ with $|\pi_2| < d$. That is, $\mathsf{depth}_g(n) < d$. Therefore, we have that $\mathsf{suc}^{g\dagger d}(n) = \mathsf{suc}^g(n)$. Since $\pi_1 \in \mathcal{P}_{g\dagger d}(n)$, this means that $\pi_1 \cdot \langle i\rangle \in \mathcal{P}(g\dagger d)$.

For the '$\supseteq$' direction, assume some $\pi \in \mathcal{P}(g\dagger d)$. By (*), π is also a position in g. To show that $\pi \in P$, let $\pi_1 < \pi$. Since only nodes of depth smaller than d can have a successor node in $g\dagger d$, the node $\mathsf{node}_{g\dagger d}(\pi_1)$ in $g\dagger d$ is at depth smaller than d. Hence, there is some $\pi_2 \sim_{g\dagger d} \pi_1$ with $|\pi_2| < d$. Because $\pi_2 \sim_{g\dagger d} \pi$ implies, by (*), that $\pi_2 \sim_g \pi$, we can conclude that $\pi \in P$.

(b) $l(\pi) = g\dagger d(\pi)$ for all $\pi \in P$. Let $\pi \in P$ and $n = \mathsf{node}_g(\pi)$. We distinguish two cases. At first, suppose that there is some $\pi' \sim_g \pi$ with $|\pi'| < d$. Then, $l(\pi) = g(\pi)$. Since $n = \mathsf{node}_g(\pi')$, we have that $\mathsf{depth}_g(n) < d$. Consequently, $\mathsf{lab}^{g\dagger d}(n) = \mathsf{lab}^g(n)$ and, therefore, $g\dagger d(\pi) = g(\pi) = l(\pi)$. In the other case that there is no $\pi' \sim_g \pi$ with $|\pi| < d$, we have $l(\pi) = \bot$. This also means that $\mathsf{depth}_g(n) = d$. Consequently, $g\dagger d(\pi) = \mathsf{lab}^{g\dagger d}(n) = \bot = l(\pi)$.

(c) $\sim\, = \,\sim_{g\dagger d}$. Using the fact that $P = \mathcal{P}(g\dagger d)$, we can conclude for all $\pi_1, \pi_2 \in P$ that

$$\begin{aligned}\pi_1 \sim_{g\dagger d} \pi_2 &\iff \mathsf{node}_{g\dagger d}(\pi_1) = \mathsf{node}_{g\dagger d}(\pi_2)\\ &\overset{(*)}{\iff} \mathsf{node}_g(\pi_1) = \mathsf{node}_g(\pi_2)\\ &\iff \pi_1 \sim_g \pi_2\\ &\iff \pi_1 \sim \pi_2\end{aligned}$$

□

Notice that a position π is retained by a truncation, i.e. $\pi \in P$, iff each node that π passes through is at a depth smaller than d (and is thus not truncated or relabelled).

From this characterisation, we immediately obtain the following relation between a term graph and its simple truncations:

Corollary 5.1. Given $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and $d \leqslant \omega$, we have the following:

(i)	$\pi \in \mathcal{P}(g)$ iff $\pi \in \mathcal{P}(g\dagger d)$	for all π with $\lvert\pi\rvert \leqslant d$.
(ii)	$g\dagger d(\pi) = g(\pi)$	for all $\pi \in \mathcal{P}(g)$ with $\lvert\pi\rvert < d$.
(iii)	$\pi_1 \sim_g \pi_2$ iff $\pi_1 \sim_{g\dagger d} \pi_2$	for all $\pi_1, \pi_2 \in \mathcal{P}(g)$ with $\lvert\pi_1\rvert, \lvert\pi_2\rvert \leqslant d$.

Proof. Using the reflexivity of $\sim_g$, (i) follows immediately from Lemma 5.2 (a). Using (i), we obtain (ii) and (iii) immediately from Lemma 5.2 (b) and (c), respectively. □

As expected, we also obtain the following relation between the simple truncation and the simple partial order.

Corollary 5.2. For each $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and $d \leqslant \omega$, we have that $g\dagger d \leqslant^{\mathsf{S}}_\bot g$.

Proof. Immediate from the characterisation of the simple truncation and the simple partial order in Lemma 5.2 and Corollary 4.1, respectively. □

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

We can now show that the metric space induced by the simple truncation is complete.

Theorem 5.1. The metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$ is complete. In particular, each Cauchy sequence $(g_\iota)_{\iota<\alpha}$ in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$ converges to the canonical term graph given by the following labelled quotient tree $(P, l, \sim)$:

$$P = \liminf_{\iota\to\alpha} \mathcal{P}(g_\iota) = \bigcup_{\beta<\alpha} \bigcap_{\beta\leqslant\iota<\alpha} \mathcal{P}(g_\iota) \qquad \sim\, = \liminf_{\iota\to\alpha} \sim_{g_\iota} = \bigcup_{\beta<\alpha} \bigcap_{\beta\leqslant\iota<\alpha} \sim_{g_\iota}$$

$$l(\pi) = g_\beta(\pi) \quad \text{for some } \beta < \alpha \text{ with } g_\iota(\pi) = g_\beta(\pi) \text{ for each } \beta \leqslant \iota < \alpha \qquad \text{for all } \pi \in P.$$

Proof. We need to check that $(P, l, \sim)$ is a well-defined labelled quotient tree. At first, we show that l is a well-defined function on P. In order to show that l is functional, assume that there are $\beta_1, \beta_2 < \alpha$ such that there is a π with $g_\iota(\pi) = g_{\beta_k}(\pi)$ for all $\beta_k \leqslant \iota < \alpha$, $k \in \{1, 2\}$. But then we have $g_{\beta_1}(\pi) = g_\beta(\pi) = g_{\beta_2}(\pi)$ for $\beta = \max\{\beta_1, \beta_2\}$.

To show that l is total on P, let $\pi \in P$ and $d = |\pi|$. By Lemma 5.1, there is some $\beta < \alpha$ such that $g_\gamma \dagger d + 1 \cong g_\iota \dagger d + 1$ for all $\beta \leqslant \gamma, \iota < \alpha$. According to Corollary 5.1, this means that all g_ι for $\beta \leqslant \iota < \alpha$ agree on positions of length smaller than $d + 1$, in particular π. Hence, $g_\iota(\pi) = g_\beta(\pi)$ for all $\beta \leqslant \iota < \alpha$, and we have $l(\pi) = g_\beta(\pi)$.

One can easily see that $\sim$ is a binary relation on P: If $\pi_1 \sim \pi_2$, then there is some $\beta < \alpha$ with $\pi_1 \sim_{g_\iota} \pi_2$ for all $\beta \leqslant \iota < \alpha$. Hence, $\pi_1, \pi_2 \in \mathcal{P}(g_\iota)$ for all $\beta \leqslant \iota < \alpha$ and thus $\pi_1, \pi_2 \in P$.

Similarly, it follows that $\sim$ is an equivalence relation on P. To show reflexivity, assume $\pi \in P$. Then there is some $\beta < \alpha$ such that $\pi \in \mathcal{P}(g_\iota)$ for all $\beta \leqslant \iota < \alpha$. Hence, $\pi \sim_{g_\iota} \pi$ for all $\beta \leqslant \iota < \alpha$ and, therefore, $\pi \sim \pi$. In the same way, symmetry and transitivity follow from the symmetry and transitivity of $\sim_{g_\iota}$.

Finally, we have to show the reachability and the congruence property from Definition 3.8. To show reachability, assume some $\pi \cdot \langle i\rangle \in P$. Then there is some $\beta < \alpha$ such that $\pi \cdot \langle i\rangle \in \mathcal{P}(g_\iota)$ for all $\beta \leqslant \iota < \alpha$. Hence, since then also $\pi \in \mathcal{P}(g_\iota)$ for all $\beta \leqslant \iota < \alpha$, we have $\pi \in P$. According to the construction of l, there is also some $\beta \leqslant \gamma < \alpha$ with $g_\gamma(\pi) = l(\pi)$. Since $\pi \cdot \langle i\rangle \in \mathcal{P}(g_\gamma)$, we can conclude that $i < \mathsf{ar}(l(\pi))$.

To establish congruence assume that $\pi_1 \sim \pi_2$. Consequently, there is some $\beta < \alpha$ such that $\pi_1 \sim_{g_\iota} \pi_2$ for all $\beta \leqslant \iota < \alpha$. Therefore, we also have for each $\beta \leqslant \iota < \alpha$ that $\pi_1 \cdot \langle i\rangle \sim_{g_\iota} \pi_2 \cdot \langle i\rangle$ for all $i < \mathsf{ar}(g_\iota(\pi_1))$ and that $g_\iota(\pi_1) = g_\iota(\pi_2)$. According to the construction of l, there is some $\beta \leqslant \gamma < \alpha$ such that $l(\pi_1) = g_\gamma(\pi_1) = g_\gamma(\pi_2) = l(\pi_2)$. Moreover, we can derive that $\pi_1 \cdot \langle i\rangle \sim \pi_2 \cdot \langle i\rangle$ for all $i < \mathsf{ar}(l(\pi_1))$.

This concludes the proof that $(P, l, \sim)$ is indeed a labelled quotient tree. Next, we show that the sequence $(g_\iota)_{\iota<\alpha}$ converges to the thus defined canonical term graph g. By Lemma 5.1, this amounts to giving for each $d < \omega$ some $\beta < \alpha$ such that $g\dagger d \cong g_\iota \dagger d$ for all $\beta \leqslant \iota < \alpha$.

To this end, let $d < \omega$. Since $(g_\iota)_{\iota<\alpha}$ is Cauchy, there is, according to Lemma 5.1, some $\beta < \alpha$ such that

$$g_\iota \dagger d \cong g_\gamma \dagger d \quad \text{for all } \beta \leqslant \iota, \gamma < \alpha. \tag{$*$}$$

In order to show that this implies that $g\dagger d \cong g_\iota \dagger d$ for all $\beta \leqslant \iota < \alpha$, we show that the respective labelled quotient trees of $g\dagger d$ and $g_\iota \dagger d$ as characterised by Lemma 5.2 coincide.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

The labelled quotient tree $(P_1, l_1, \sim_1)$ for $g \dagger d$ is given by

$$P_1 = \{\pi \in P \mid \forall \pi_1 < \pi \exists \pi_2 \sim \pi_1 : |\pi_2| < d\} \qquad l_1(\pi) = \begin{cases} l(\pi) & \text{if } \exists \pi' \sim \pi : |\pi'| < d \\ \bot & \text{otherwise} \end{cases}$$
$$\sim_1 \; = \sim \cap \, P_1 \times P_1$$

The labelled quotient tree $(P_2^\iota, l_2^\iota, \sim_2^\iota)$ for each $g_\iota \dagger d$ is given by

$$P_2^\iota = \{\pi \in \mathcal{P}(g_\iota) \mid \forall \pi_1 < \pi \exists \pi_2 \sim_{g_\iota} \pi_1 : |\pi_2| < d\} \qquad \sim_2^\iota = \sim_{g_\iota} \cap \, P_2^\iota \times P_2^\iota$$
$$l_2^\iota(\pi) = \begin{cases} g_\iota(\pi) & \text{if } \exists \pi' \sim_{g_\iota} \pi : |\pi'| < d \\ \bot & \text{otherwise} \end{cases}$$

Due to $(*)$, all $(P_2^\iota, l_2^\iota, \sim_2^\iota)$ with $\beta \leqslant \iota < \alpha$ are pairwise equal. Therefore, we write $(P_2, l_2, \sim_2)$ for this common labelled quotient tree. That is, it remains to be shown that $(P_1, l_1, \sim_1)$ and $(P_2, l_2, \sim_2)$ are equal.

(a) $P_1 = P_2$. For the '$\subseteq$' direction, let $\pi \in P_1$. If $\pi = \langle\rangle$, we immediately have that $\pi \in P_2$. Hence, we can assume that π is non-empty. Since $\pi \in P_1$ implies $\pi \in P$, there is some $\beta \leqslant \beta' < \alpha$ with $\pi \in \mathcal{P}(g_\iota)$ for all $\beta' \leqslant \iota < \alpha$. Moreover, this means that for each $\pi_1 < \pi$, there is some $\pi_2 \sim \pi_1$ with $|\pi_2| < d$. That is, there is some $\beta' \leqslant \gamma_{\pi_1} < \alpha$ such that $\pi_2 \sim_{g_\iota} \pi_1$ for all $\gamma_{\pi_1} \leqslant \iota < \alpha$. Since there are only finitely many proper prefixes $\pi_1 < \pi$ but at least one, we can define $\gamma = \max\{\gamma_{\pi_1} \mid \pi_1 < \pi\}$ such that we have for each $\pi_1 < \pi$ some $\pi_2 \sim_{g_\gamma} \pi_1$ with $|\pi_2| < d$. Hence, $\pi \in P_2^\gamma = P_2$.

To show the converse direction, assume that $\pi \in P_2$. Then, $\pi \in P_2^\iota \subseteq \mathcal{P}(g_\iota)$ for all $\beta \leqslant \iota < \alpha$. Hence, $\pi \in P$. To show that $\pi \in P_1$, assume some $\pi_1 < \pi$. Since $\pi \in P_2^\beta$, there is some $\pi_2 \sim_{g_\beta} \pi_1$ with $|\pi_2| < d$. Then, $\pi_1 \in P_2$ because P_2 is closed under prefixes and $\pi_2 \in P_2$ because $|\pi_2| < d$. Thus, $\pi_2 \sim_2 \pi_1$ which implies $\pi_2 \sim_{g_\iota} \pi_1$ for all $\beta \leqslant \iota < \alpha$. Consequently, $\pi_2 \sim \pi_1$, which means that $\pi \in P_1$.

(c) $\sim_1 = \sim_2$. For the '$\subseteq$' direction, assume $\pi_1 \sim_1 \pi_2$. Hence, $\pi_1 \sim \pi_2$ and $\pi_1, \pi_2 \in P_1 = P_2$. This means that there is some $\beta \leqslant \gamma < \alpha$ with $\pi_1 \sim_{g_\gamma} \pi_2$. Consequently, $\pi_1 \sim_2 \pi_2$. For the converse direction, assume that $\pi_1 \sim_2 \pi_2$. Then $\pi_1, \pi_2 \in P_2 = P_1$ and $\pi_1 \sim_{g_\iota} \pi_2$ for all $\beta \leqslant \iota < \alpha$. Hence, $\pi_1 \sim \pi_2$ and we can conclude that $\pi_1 \sim_1 \pi_2$.

(b) $l_1 = l_2$. We show this by proving that, for all $\beta \leqslant \iota < \alpha$, the condition $\exists \pi' \sim \pi : |\pi'| < d$ from the definition of l_1 is equivalent to the condition $\exists \pi' \sim_{g_\iota} \pi : |\pi'| < d$ from the definition of l_2 and that $l(\pi) = g_\iota(\pi)$ if either condition is satisfied. The latter is simple: Whenever there is some $\pi' \sim \pi$ with $|\pi'| < d$, then $g_\iota(\pi) = l_2^\iota(\pi) = l_2^\beta(\pi) = g_\beta(\pi)$ for all $\beta \leqslant \iota < \alpha$. Hence, $l(\pi) = g_\beta(\pi) = g_\iota(\pi)$ for all $\beta \leqslant \iota < \alpha$. For the former, we first consider the 'only if' direction of the equivalence. Let $\pi \in P_1$ and $\pi' \sim \pi$ with $|\pi'| < d$. Then also $\pi' \in P_1$ which means that $\pi' \sim_1 \pi$. Since then $\pi' \sim_2 \pi$, we can conclude that $\pi' \sim_{g_\iota} \pi$ for all $\beta \leqslant \iota < \alpha$. For the converse direction, assume that $\pi \in P_2$, $\pi' \sim_{g_\iota} \pi$ and $|\pi'| < d$. Then also $\pi' \in P_2$ which means that $\pi' \sim_2 \pi$. This implies $\pi' \sim_1 \pi$, which in turn implies $\pi' \sim \pi$. □

Example 5.1. Reconsider the two sequences of term graphs $(g_\iota)_{\iota<\omega}$ and $(h_\iota)_{\iota<\omega}$ from Figure 6c and 6d on page 40. The simple truncation of the term graphs g_ι at depth 2 alternates between the term trees $a :: \bot :: \bot$ and $b :: \bot :: \bot$. More precisely, $g_\iota \dagger 2 = a :: \bot :: \bot$

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

if ι is even and $g_\iota \dagger 2 = b :: \bot :: \bot$ if ι is odd. According to Lemma 5.1, this means that $(g_\iota)_{\iota<\omega}$ is not Cauchy in $(\mathcal{T}^\infty(\Sigma), \mathbf{d}_\dagger)$ and is consequently not convergent.

On the other hand, $(h_\iota)_{\iota<\omega}$ does converge to the term graph h_ω in $(\mathcal{T}^\infty(\Sigma), \mathbf{d}_\dagger)$: For each $d \in \mathbb{N}$, we have that $h_\omega \dagger d+1 \cong h_\iota \dagger d+1$ for all $d \leqslant \iota < \omega$. Lemma 5.1 then yields that $\lim_{\iota\to\omega} h_\iota = h_\omega$.

As we have seen in Example 4.1, the limit inferior induced by $\leqslant^{\mathsf{S}}_\bot$ showed some curious behaviour for the sequence of term graphs illustrated in Figure 3. This is not the case for the metric $\mathbf{d}_\dagger$. In fact, there is no topological space in which $(g_\iota)_{\iota<\omega}$ from Figure 3 converges to a unique limit. In particular, this means that there is no metric space in which $(g_\iota)_{\iota<\omega}$ converges.

5.3. *Other truncation functions and their metric spaces*

Generalising concepts from terms to term graphs is not a straightforward matter as we have to decide how to deal with additional sharing that term graphs offer. The definition of simple truncation seems to be an obvious choice for a generalisation of tree truncation. In this section, we shall formally argue that it is in fact the case. More specifically, we show that no matter how we define the sharing of the $\bot$-nodes that fill the holes caused by the truncation, we obtain the same topology.

The following lemma is a handy tool for comparing metric spaces induced by truncation functions:

Lemma 5.3. Let τ, υ be two truncation functions on $\mathcal{G}^\infty(\Sigma_\bot)$ and $f\colon \mathcal{G}^\infty_\mathcal{C}(\Sigma) \to \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ a function on $\mathcal{G}^\infty_\mathcal{C}(\Sigma)$. Then the following are equivalent:

(i) f is a continuous mapping $f\colon (\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau) \to (\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\upsilon)$.

(ii) For each $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ and $d < \omega$, there is some $e < \omega$ such that

$$\mathsf{sim}_\tau(g,h) \geqslant e \quad\Longrightarrow\quad \mathsf{sim}_\upsilon(f(g), f(h)) \geqslant d \quad \text{for all } h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma).$$

(iii) For each $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ and $d < \omega$, there is some $e < \omega$ such that

$$\tau_e(g) \cong \tau_e(h) \quad\Longrightarrow\quad \upsilon_d(f(g)) \cong \upsilon_d(f(h)) \quad \text{for all } h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma).$$

Proof. Analogous to Lemma 5.1. □

An easy consequence of the above lemma is that if two truncation functions only differ by a constant depth, they induce the same topology.

Proposition 5.3. Let τ, υ be two truncation functions on $\mathcal{G}^\infty(\Sigma_\bot)$ such that there is a $\delta < \omega$ with $|\mathsf{sim}_\tau(g,h) - \mathsf{sim}_\upsilon(g,h)| \leqslant \delta$ for all $g, h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$. Then, $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau)$ and $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\upsilon)$ are topologically equivalent, i.e. they induce the same topology.

Proof. We show that the identity function $\mathsf{id}\colon \mathcal{G}^\infty_\mathcal{C}(\Sigma) \to \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ is a homeomorphism from $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau)$ to $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\upsilon)$, i.e. both id and id^{-1} are continuous. Due to the symmetry of the setting, it suffices to show that id is continuous. To this end, let $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ and $d < \omega$. Define $e = d + \delta$ and assume some $h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ such that $\mathsf{sim}_\tau(g,h) \geqslant e$.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

By Lemma 5.3, it remains to be shown that then $\mathsf{sim}_\upsilon(g,h) \geqslant d$. Indeed, we have $\mathsf{sim}_\upsilon(g,h) \geqslant \mathsf{sim}_\tau(g,h) - \delta \geqslant e - \delta = d$. □

This shows that metric spaces induced by truncation functions are essentially invariant under changes in the truncation function bounded by a constant margin.

Remark 5.1. We should point out that the original definition of the metric on terms by Arnold and Nivat (1980) was slightly different from the one we showed here. Recall that we defined similarity as the maximum depth of truncation that ensures equality:

$$\mathsf{sim}_\tau(g,h) = \max\{d \leqslant \omega \mid \tau_d(g) \cong \tau_d(h)\}.$$

Arnold and Nivat, on the other hand, defined it as the minimum truncation depth that still shows inequality:

$$\mathsf{sim}'_\tau(g,h) = \min\{d \leqslant \omega \mid \tau_d(g) \ncong \tau_d(h)\}.$$

However, it is easy to see that either both $\mathsf{sim}_\tau(g,h)$ and $\mathsf{sim}'_\tau(g,h)$ are ω or $\mathsf{sim}'_\tau(g,h) = \mathsf{sim}_\tau(g,h) + 1$. Hence, by Proposition 5.3, both definitions yield the same topology.

Proposition 5.3 also shows that two truncation functions induce the same topology if they only differ in way they treat 'fringe nodes,' i.e. nodes that are introduced in place of the nodes that have been cut off. Since the definition of truncation functions requires that $\tau_0(g) \cong \bot$ and $\tau_\omega(g) \cong g$, we do not give the explicit construction of the truncation for the depths 0 and ω in the examples below.

The truncation of term graphs in general – as opposed to the truncation of term trees – has some peculiar effects: Nodes in a term graph may not be cut off by a truncation even though some nodes on *a* path to them are cut off; thus, the sharing of nodes that are retained in a truncation may be altered.

Definition 5.4 (Acyclic predecessors). Given $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and $n, m \in N^g$, we say that m is an *acyclic predecessor* of n in g if there is an acyclic position $\pi \cdot \langle i \rangle \in \mathcal{P}^a_g(n)$ with $\pi \in \mathcal{P}_g(m)$. The set of acyclic predecessors of n in g is denoted $\mathsf{Pre}^a_g(n)$.

The distinction between acyclic and cyclic predecessors will play a prominent role in the truncation functions that we shall discuss below.

Example 5.2. Consider the following variant τ of the simple truncation function †. Let $g \in \mathcal{G}^\infty(\Sigma_\bot)$ be a term graph. For each $n \in N^g$ and $i \in \mathbb{N}$, we use n^i to denote a fresh node, i.e. $\left\{n^i \,\middle|\, n \in N^g, i \in \mathbb{N}\right\}$ is a set of pairwise distinct nodes not occurring in N^g. Given a depth $0 < d < \omega$, we define the truncation $\tau_d(g)$ as follows:

$$N^{\tau_d(g)} = N^g_{<d} \uplus N^g_{=d}$$
$$N^g_{<d} = \{n \in N^g \mid \mathsf{depth}_g(n) < d\}$$
$$N^g_{=d} = \left\{n^i \,\middle|\, n \in N^g_{<d}, 0 \leqslant i < \mathsf{ar}_g(n), \mathsf{suc}^g_i(n) \notin N^g_{<d}\right\}$$
$$\mathsf{lab}^{\tau_d(g)}(n) = \begin{cases} \mathsf{lab}^g(n) & \text{if } n \in N^g_{<d} \\ \bot & \text{if } n \in N^g_{=d} \end{cases} \qquad \mathsf{suc}^{\tau_d(g)}_i(n) = \begin{cases} \mathsf{suc}^g_i(n) & \text{if } n^i \notin N^g_{=d} \\ n^i & \text{if } n^i \in N^g_{=d} \end{cases}$$

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

One can easily show that τ is in fact a truncation function. The difference between $\dagger$ and τ is that in the latter we create a fresh node n^i whenever a node n has a successor $\mathsf{suc}_i^g(n)$ that lies at the fringe, i.e. at depth d. Since this only affects the nodes at the fringe and, therefore, only nodes at the same depth d we get the following:

$$g\dagger d \cong h\dagger d \quad\Longrightarrow\quad \tau_d(g) \cong \tau_d(h), \text{ and}$$
$$\tau_d(g) \cong \tau_d(h) \quad\Longrightarrow\quad g\dagger d-1 \cong h\dagger d-1.$$

Hence, the respectively induced similarities only differ by a constant margin of 1, i.e. we have that $|\mathsf{sim}_\dagger(g,h) - \mathsf{sim}_\tau(g,h)| = 1$. According to Proposition 5.3, this means that $(\mathcal{G}_\mathcal{C}^\infty(\Sigma), \mathbf{d}_\dagger)$ and $(\mathcal{G}_\mathcal{C}^\infty(\Sigma), \mathbf{d}_\tau)$ are topologically equivalent.

Consider another variant υ of the simple truncation function $\dagger$. Given a term graph $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and depth $0 < d < \omega$, we define the truncation $\upsilon_d(g)$ as follows:

$$N^{\upsilon_d(g)} = N^g_{<d} \uplus N^g_{=d}$$
$$N^g_{<d} = \{n \in N^g \mid \mathsf{depth}_g(n) < d\}$$
$$N^g_{=d} = \left\{ n^i \;\middle|\; \begin{array}{rl} n \in N^g, \mathsf{depth}_g(n) = d-1, 0 \leqslant i < \mathsf{ar}_g(n) \text{ with} & \mathsf{suc}_i^g(n) \notin N^g_{<d} \\ \text{or} & n \notin \mathsf{Pre}_g^a(\mathsf{suc}_i^g(n)) \end{array} \right\}$$
$$\mathsf{lab}^{\upsilon_d(g)}(n) = \begin{cases} \mathsf{lab}^g(n) & \text{if } n \in N^g_{<d} \\ \bot & \text{if } n \in N^g_{=d} \end{cases} \qquad \mathsf{suc}^{\upsilon_d(g)}(n) = \begin{cases} \mathsf{suc}_i^g(n) & \text{if } n^i \notin N^g_{=d} \\ n^i & \text{if } n^i \in N^g_{=d} \end{cases}$$

Also υ forms a truncation function as one can easily show. In addition to creating fresh nodes n^i for each successor that is not in the retained nodes $N^g_{<d}$, the truncation function υ creates such new nodes n^i for each cycle that is created by a node just above the fringe. Again, as for the truncation function τ, only the nodes at the fringe, i.e. at depth d are affected by this change. Hence, the respectively induced similarities of $\dagger$ and υ only differ by a constant margin of 1, which makes the metric spaces $(\mathcal{G}_\mathcal{C}^\infty(\Sigma), \mathbf{d}_\dagger)$ and $(\mathcal{G}_\mathcal{C}^\infty(\Sigma), \mathbf{d}_\upsilon)$ topologically equivalent as well.

The robustness of the metric space $(\mathcal{G}_\mathcal{C}^\infty(\Sigma), \mathbf{d}_\dagger)$ under the changes illustrated above is due to the uniformity of the core definition of the simple truncation which only takes into account the depth. By simply increasing the depth by a constant number, we can compensate for changes in the way fringe nodes are dealt with.

This is much different for the rigid truncation function $g\ddagger d$ that we have used in our previous work (Bahr 2012b) in order to derive a complete metric on term graph:

Definition 5.5 (Rigid truncation of term graphs). Let $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and $d \in \mathbb{N}$.

(i) The set of *retained nodes* of g at d, denoted $N^g_{<d}$, is the least subset M of N^g satisfying the following conditions for all $n \in N^g$:

$$\text{(T1) } \mathsf{depth}_g(n) < d \implies n \in M \qquad \text{(T2) } n \in M \implies \mathsf{Pre}_g^a(n) \subseteq M.$$

(ii) For each $n \in N^g$ and $i \in \mathbb{N}$, we use n^i to denote a fresh node, i.e. $\{n^i \mid n \in N^g, i \in \mathbb{N}\}$ is a set of pairwise distinct nodes not occurring in N^g. The set of *fringe nodes* of g at

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

d, denoted $N^g_{=d}$, is defined as the singleton set $\{r^g\}$ if $d = 0$, and otherwise as the set

$$\left\{ n^i \;\middle|\; \begin{array}{rl} n \in N^g_{<d}, 0 \leqslant i < \mathsf{ar}_g(n) \text{ with} & \mathsf{suc}^g_i(n) \notin N^g_{<d} \\ \text{or} & \mathsf{depth}_g(n) \geqslant d-1, n \notin \mathsf{Pre}^a_g(\mathsf{suc}^g_i(n)) \end{array} \right\}$$

(iii) The *rigid truncation* of g at d, denoted $g\ddagger d$, is the term graph defined by

$$N^{g\ddagger d} = N^g_{<d} \uplus N^g_{=d} \qquad r^{g\ddagger d} = r^g$$

$$\mathsf{lab}^{g\ddagger d}(n) = \begin{cases} \mathsf{lab}^g(n) & \text{if } n \in N^g_{<d} \\ \bot & \text{if } n \in N^g_{=d} \end{cases} \qquad \mathsf{suc}^{g\ddagger d}_i(n) = \begin{cases} \mathsf{suc}^g_i(n) & \text{if } n^i \notin N^g_{=d} \\ n^i & \text{if } n^i \in N^g_{=d} \end{cases}$$

Additionally, we define $g\ddagger\omega$ to be the term graph g itself.

The idea of this definition of truncation is that not only each node at depth $< d$ is kept – via the closure condition (T1) – but also every acyclic predecessor of such a node – via (T2). In sum, every node on an acyclic path from the root to a node at depth smaller than d is kept. The difference between the two truncation functions † and ‡ are illustrated in Figure 4.

In contrast to the simple truncation †, the rigid truncation function ‡ is quite vulnerable to small changes:

Example 5.3. Consider the following variant τ of the rigid truncation function ‡. Given a term graph $g \in \mathcal{G}^\infty(\Sigma_\bot)$ and depth $d \in \mathbb{N}^+$, we define the truncation $\tau_d(g)$ as follows: the set of retained nodes $N^g_{<d}$ is defined as for the truncation $g\ddagger d$. For the rest, we define

$$N^g_{=d} = \left\{ \mathsf{suc}^g_i(n) \,\middle|\, n \in N^g_{<d}, 0 \leqslant i < \mathsf{ar}_g(n), \mathsf{suc}^g_i(n) \notin N^g_{<d} \right\}$$
$$N^{\tau_d(g)} = N^g_{<d} \uplus N^g_{=d}$$
$$\mathsf{lab}^{\tau_d(g)}(n) = \begin{cases} \mathsf{lab}^g(n) & \text{if } n \in N^g_{<d} \\ \bot & \text{if } n \in N^g_{=d} \end{cases} \qquad \mathsf{suc}^{\tau_d(g)}(n) = \begin{cases} \mathsf{suc}^g(n) & \text{if } n \in N^g_{<d} \\ \langle\rangle & \text{if } n \in N^g_{=d} \end{cases}$$

In this variant of truncation, some sharing of the retained nodes is preserved. Instead of creating fresh nodes for each successor node that is not in the set of retained nodes, we simply keep the successor node. Additionally, loops back into the retained nodes are not cut off. This variant of the truncation deals with its retained nodes in essentially the same way as the simple truncation. However, opposed the simple truncation and their variants, this truncation function yields a topology different from the metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\ddagger)$! To see this, consider the two families of term graphs g_n and h_n illustrated in Figure 5. For both families, we have that the τ-truncations at depth 2 to $n+2$ are the same, i.e. $\tau_d(g_n) = \tau_2(g_n)$ and $\tau_d(h_n) = \tau_2(h_n)$ for all $2 \leqslant d \leqslant n+2$. The same holds for the truncation function ‡. Moreover, since the two leftmost successors of the h-node are not shared in g_n, both truncation functions coincide on g_n, i.e. $g_n\ddagger d = \tau_d(g_n)$. This is not the case for h_n. In fact, they only coincide up to depth 1. In total, we can observe that $\mathsf{sim}_\ddagger(g_n, h_n) = n+2$, but $\mathsf{sim}_\tau(g_n, h_n) = 1$. This means, however, that the sequence $\langle g_0, h_0, g_1, h_1, \ldots\rangle$ converges in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\ddagger)$ but not in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\tau)$!

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

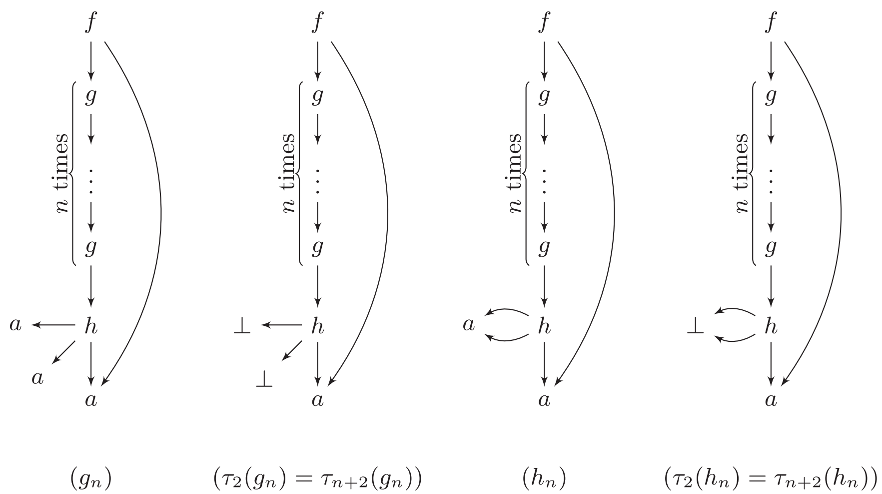


Fig. 5. Variations in fringe nodes.

A similar example can be constructed that uses the difference in the way the two truncation functions deal with fringe nodes created by cycles back into the set of retained nodes.

The above discussion should give a first indication why the simple metric $\mathbf{d}_\dagger$ should be preferred over the rigid partial order $\mathbf{d}_\ddagger$: the metric $\mathbf{d}_\dagger$ is not only simpler than $\mathbf{d}_\ddagger$ but also more natural in the sense that we obtain the topology of the metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$ without paying too much attention to the corner case details of the underlying truncation function. Small changes in the way we treat these corner cases do not affect the resulting topology as we have illustrated in Example 5.2. For the metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\ddagger)$, on the other hand, we have to be very careful about how to deal with fringe nodes. As Example 5.3 shows, even small changes yield a different topology. This is part of the reason why the definition of the underlying rigid truncation ‡ is so convoluted.

In Section 8.3, we will give another reason to prefer the metric $\mathbf{d}_\dagger$ over the metric $\mathbf{d}_\ddagger$: While the former allows us to construct the set of term graphs from the set of finite term graphs via metric completion, the latter does not. That is, the rigid metric does not yield a representation of infinite term graphs as the limit of a sequence of finite term graphs.

6. Infinitary term graph rewriting

In the previous sections, we have constructed and investigated the necessary metric and partial order structures upon which the infinitary calculus of term graph rewriting that we shall introduce in this section is based. After describing the framework of term graph rewriting that we consider, we will explore different modes of convergence on term graphs. In the same way that infinitary term rewriting instantiates the abstract notions of weak *m*- and *p*-convergence (Bahr 2010), infinitary term graph rewriting is an instantiation of these abstract modes of convergence to term graphs.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

6.1. *Term graph rewriting systems*

We base our infinitary term rewriting calculus on the term graph rewriting framework of Barendregt et al. (1987). In order to represent placeholders in rewrite rules, this framework uses variables – in a manner much similar to term rewrite rules. However, instead of open graphs whose unlabelled nodes are interpreted as variables, we use explicit variable symbols. To this end, we consider a signature $\Sigma_{\mathcal{V}} = \Sigma \uplus \mathcal{V}$ that extends the signature Σ with a set $\mathcal{V}$ of nullary variable symbols.

Definition 6.1 (Term graph rewriting system).

(i) Given a signature Σ, a *term graph rule* ρ over Σ is a triple (g, l, r) where g is a graph over $\Sigma_{\mathcal{V}}$ and $l, r \in N^g$, such that all nodes in g reachable from l or r. We write ρ_l and ρ_r to denote the left- and right-hand side of ρ, respectively, i.e. the term graphs $g|_l$ and $g|_r$. Additionally, we require that, for each variable $v \in \mathcal{V}$, there is at most one node n in g labelled v and n is different but still reachable from l.

(ii) A term graph rewriting system (GRS) $\mathcal{R}$ is a pair (Σ, R) with Σ a signature and R a set of term graph rules over Σ.

The requirement that the root l of the left-hand side is not labelled with a variable symbol is analogous to the requirement that the left-hand side of a term rule is not a variable. Similarly, the restriction that nodes labelled with variable symbols must be reachable from the root of the left-hand side corresponds to the restriction on term rules that every variable occurring on the right-hand side must also occur on the left-hand side.

Term graphs can be used to compactly represent terms. This representation of terms is defined by the unravelling of term graphs. This notion can be extended to term graph rules.

Definition 6.2 (Unravelling of term graph rules). Let ρ be a term graph rule with ρ_l the left- and ρ_r the right-hand side term graph. The *unravelling* of ρ, denoted $\mathcal{U}(\rho)$, is the term rule $\mathcal{U}(\rho_l) \to \mathcal{U}(\rho_r)$. Let $\mathcal{R} = (\Sigma, R)$ be a GRS. The unravelling of $\mathcal{R}$, denoted $\mathcal{U}(\mathcal{R})$, is the TRS $(\Sigma, \mathcal{U}(R))$ with $\mathcal{U}(R) = \{\mathcal{U}(\rho) \mid \rho \in G\}$.

Figure 6a illustrates two term graph rules that both represent the term rule $x :: y :: z \to y :: x :: y :: z$ from Example 2.1, which they unravel to.

The application of a rewrite rule ρ (with root nodes l and r) to a term graph g is performed in four steps: At first, a suitable subterm graph of g rooted in some node n of g is *matched* against the left-hand side of ρ. This amounts to finding a $\mathcal{V}$-homomorphism $\phi\colon \rho_l \to_{\mathcal{V}} g|_n$ from the term graph rooted in l to the subterm graph rooted in n, the *redex*. The $\mathcal{V}$-homomorphism ϕ allows to instantiate variables in the rule with subterm graphs of the redex. In the second step, nodes and edges in ρ that are not reachable from l are copied into g, such that edges pointing to nodes in the term graph rooted in l are redirected to the image under ϕ. In the last two steps, all edges pointing to n are redirected to (the copy of) r and all nodes not reachable from the root of (the now modified version of) g are removed.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

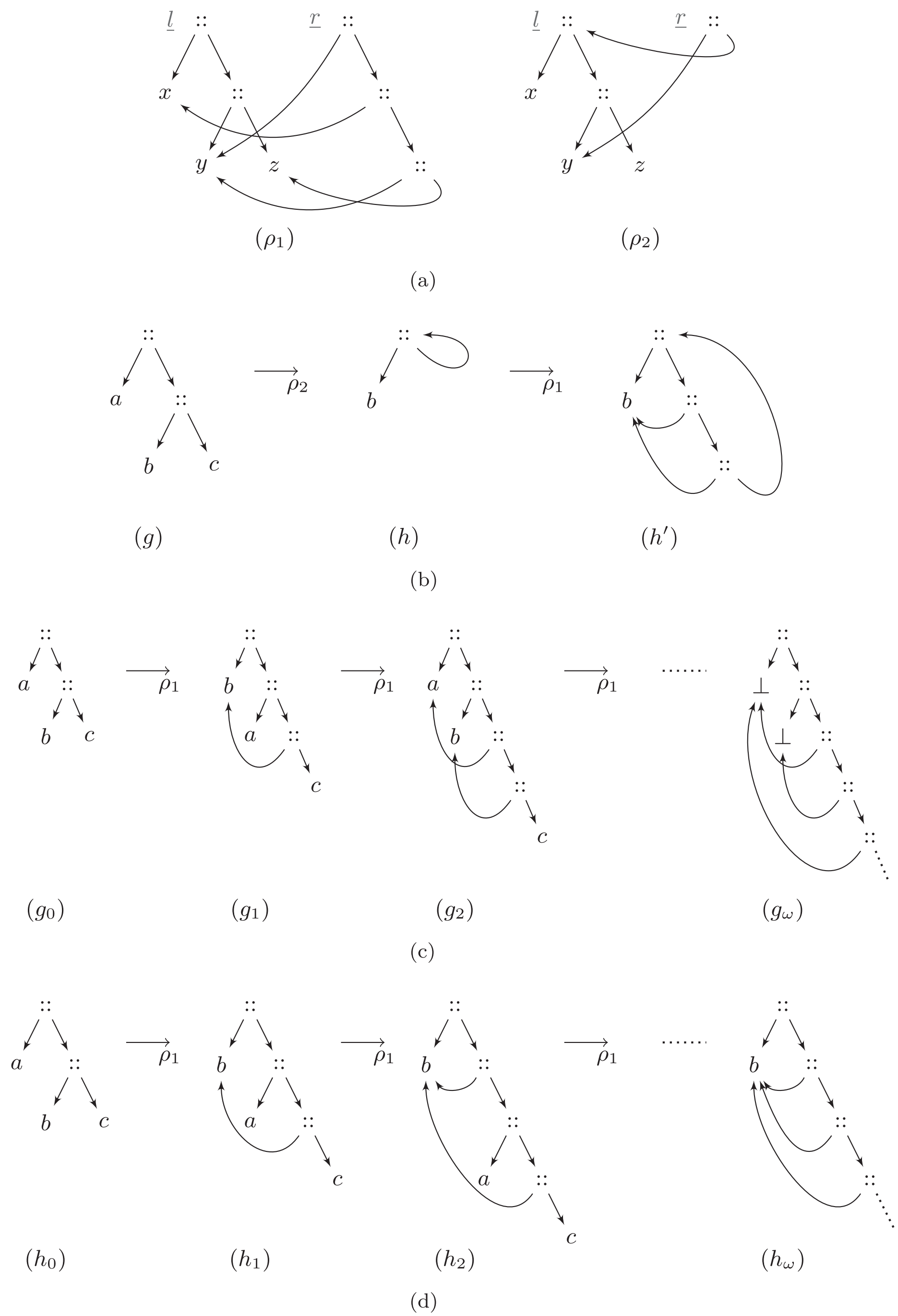


Fig. 6. Term graph rules and their reductions. (a) Term graph rules that unravel to $x :: y :: z \to y :: x :: y :: z$. (b) A ρ_2-step followed by a ρ_1-step. (c) A term graph reduction over ρ_1 that does not weakly m-converge. (d) A weakly m-converging term graph reduction over ρ_1.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Definition 6.3 (Application of a term graph rewrite rule, Barendregt et al. (1987)). Let $\rho = (N^\rho, \mathsf{lab}^\rho, \mathsf{suc}^\rho, l^\rho, r^\rho)$ be a term graph rewrite rule in a GRS $\mathcal{R} = (\Sigma, R)$, $g \in \mathcal{G}^\infty(\Sigma)$ and $n \in N^g$. ρ is called *applicable* to g at n if there is a $\mathcal{V}$-homomorphism $\phi\colon \rho_l \to_\mathcal{V} g|_n$. ϕ is called the *matching $\mathcal{V}$-homomorphism* of the rule application, and $g|_n$ is called a *ρ-redex*. Next, we define the *result* of the application of the rule ρ to g at n using the $\mathcal{V}$-homomorphism ϕ. This is done by constructing the intermediate graphs g_1 and g_2, and the final result g_3.

(i) The graph g_1 is obtained from g by adding the part of ρ not contained in the left-hand side:

$$N^{g_1} = N^g \uplus (N^\rho \setminus N^{\rho_l})$$

$$\mathsf{lab}^{g_1}(m) = \begin{cases} \mathsf{lab}^g(m) & \text{if } m \in N^g \\ \mathsf{lab}^\rho(m) & \text{if } m \in N^\rho \setminus N^{\rho_l} \end{cases}$$

$$\mathsf{suc}_i^{g_1}(m) = \begin{cases} \mathsf{suc}_i^g(m) & \text{if } m \in N^g \\ \mathsf{suc}_i^\rho(m) & \text{if } m, \mathsf{suc}_i^\rho(m) \in N^\rho \setminus N^{\rho_l} \\ \phi(\mathsf{suc}_i^\rho(m)) & \text{if } m \in N^\rho \setminus N^{\rho_l}, \mathsf{suc}_i^\rho(m) \in N^{\rho_l}. \end{cases}$$

(ii) Let $n' = \phi(r^\rho)$ if $r^\rho \in N^{\rho_l}$ and $n' = r^\rho$ otherwise. The graph g_2 is obtained from g_1 by redirecting edges ending in n to n':

$$N^{g_2} = N^{g_1} \qquad \mathsf{lab}^{g_2} = \mathsf{lab}^{g_1} \qquad \mathsf{suc}_i^{g_2}(m) = \begin{cases} \mathsf{suc}_i^{g_1}(m) & \text{if } \mathsf{suc}_i^{g_1}(m) \neq n \\ n' & \text{if } \mathsf{suc}_i^{g_1}(m) = n. \end{cases}$$

(iii) The term graph g_3 is obtained by setting the root node r', which is r if $l = r^g$, and otherwise r^g. That is, $g_3 = g_2|_{r'}$. This also means that all nodes not reachable from r' in g_2 are removed.

The above construction induces a *pre-reduction step* $\psi = (g, n, \rho, n', g_3)$ from g to g_3, written $\psi\colon g \mapsto_{n,\rho,n'} g_3$. In order to indicate the underlying GRS $\mathcal{R}$, we also write $\psi\colon g \mapsto_\mathcal{R} g_3$.

Examples for term graph (pre-)reduction steps are shown in Figure 6. We revisit them in more detail in Example 6.1 in the next section.

Note that term graph rules do not provide a duplication mechanism. Each variable is allowed to occur at most once. Duplication must always be simulated by sharing, i.e. with nodes reachable via multiple paths from any of the two roots. This means for example that a variable that should 'occur' on the left- and the right-hand side must be shared between the left- and the right-hand side of the rule as seen in the term graph rules in Figure 6a. This sharing can be direct as in ρ_1 – the variable node has multiple ingoing edges – or indirect as in ρ_2 – the variable node is reachable from nodes with multiple ingoing edges. Likewise, for variables that are supposed to be duplicated on the right-hand side, e.g. the variable y in the term rule $x :: y :: z \to y :: x :: y :: z$, we have to use sharing in order to represent multiple occurrence of the same variable as seen in the corresponding term graph rules in Figure 6a: In both rules, the y-node is reachable by two distinct paths from the right-hand side root r.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

The definition of term graph rewriting in the form of pre-reduction steps is very operational in style. The result of applying a rewrite rule to a term graph is constructed in several steps by manipulating nodes and edges explicitly. While this is beneficial for implementing a rewriting system, this is problematic for reasoning on term graphs up to isomorphisms, which is necessary for introducing notions of convergence. In our case, however, this does not cause any harm since the construction in Definition 6.3 is invariant under isomorphism:

Proposition 6.1 (Pre-reduction steps). Let $\phi\colon g \mapsto_{n,\rho,m} h$ be a pre-reduction step in some GRS $\mathcal{R}$ and $\psi_1\colon g' \cong g$. Then there is a pre-reduction step $\phi'\colon g' \mapsto_{n',\rho,m'} h'$ with $\psi_2\colon h' \cong h$ such that $\psi_1(n') = n$ and $\psi_1(m') = m$.

Proof. Immediate from the construction in Definition 6.3. □

This justifies the following definition of reduction steps:

Definition 6.4 (Reduction steps). Let $\mathcal{R} = (\Sigma, R)$ be a GRS, $\rho \in R$ and $g, h \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ with $n \in N^g$ and $m \in N^h$. A tuple $\phi = (g, n, \rho, m, h)$ is called a *reduction step*, written $\phi\colon g \to_{n,\rho,m} h$, if there is a pre-reduction step $\phi'\colon g' \to_{n',\rho,m'} h'$ with $\mathcal{C}(g') = g$, $\mathcal{C}(h') = h$, $n = \mathcal{P}_{g'}(n')$, and $m = \mathcal{P}_{h'}(m')$. As for pre-reduction step, we also write $\phi\colon g \to_\mathcal{R} h$ or simply $\phi\colon g \to h$ for short.

In other words, a reduction step is a canonicalised pre-reduction step.

6.2. *Convergence of transfinite reductions*

In this section, we shall look at term graph reductions of potentially transfinite length.

Definition 6.5 (Reduction). Let $\mathcal{R} = (\Sigma, R)$ be a GRS. A *reduction* in $\mathcal{R}$ is a sequence $(g_\iota \to_\mathcal{R} g_{\iota+1})_{\iota<\alpha}$ of rewriting steps in $\mathcal{R}$. If S is finite, we write $S\colon g_0 \to^* g_\alpha$.

In analogy to infinitary term rewriting, we employ the partial order $\leqslant^{\mathsf{S}}_\bot$ and the metric $\mathbf{d}_\dagger$ for the purpose of defining convergence of transfinite term graph reductions.

Definition 6.6 (Convergence of reductions). Let $\mathcal{R} = (\Sigma, R)$ be a GRS.

(i) Let $S = (g_\iota \to_\mathcal{R} g_{\iota+1})_{\iota<\alpha}$ be a reduction in $\mathcal{R}$. S is *weakly m-continuous*, written $S\colon g_0 \xrightarrow{m}_\mathcal{R} \dots$, if the underlying sequence of term graphs $(g_\iota)_{\iota<\widehat{\alpha}}$ is continuous, i.e. $\lim_{\iota\to\lambda} g_\iota = g_\lambda$ for each limit ordinal $\lambda < \alpha$. S *weakly m-converges* to $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ in $\mathcal{R}$, written $S\colon g_0 \xrightarrow{m}_\mathcal{R} g$, if it is weakly m-continuous and $\lim_{\iota\to\widehat{\alpha}} g_\iota = g$.

(ii) Let $\mathcal{R}_\bot$ be the GRS $(\Sigma_\bot, R)$ over the extended signature $\Sigma_\bot$ and $S = (g_\iota \to_{\mathcal{R}_\bot} g_{\iota+1})_{\iota<\alpha}$ a reduction in $\mathcal{R}_\bot$. S is *weakly p-continuous*, written $S\colon g_0 \xrightarrow{p}_\mathcal{R} \dots$, if $\liminf_{\iota<\lambda} g_i = g_\lambda$ for each limit ordinal $\lambda < \alpha$. S *weakly p-converges* to $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$ in $\mathcal{R}$, written $S\colon g_0 \xrightarrow{p}_\mathcal{R} g$, if it is weakly p-continuous and $\liminf_{\iota<\widehat{\alpha}} g_i = g$.

Note that we have to extend the signature of $\mathcal{R}$ to $\Sigma_\bot$ for the definition of weak p-convergence. Moreover, since the partial order $\leqslant^{\mathsf{S}}_\bot$ forms a complete semilattice on $\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$, weak p-continuity coincides with weak p-convergence

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Example 6.1. Figure 6a shows two term graph rules that both unravel to the term rule $x :: y :: z \to y :: x :: y :: z$ from Example 2.1. The two rules differ only in their sharing with ρ_1 using 'minimal sharing' and ρ_2 using 'maximal sharing.'

Figure 6c and d illustrate term graph reductions that correspond to the term reductions T and T' from Example 2.1 and 2.2, respectively. All reductions – including the term graph reductions – start from the same term (tree) $a :: b :: c$.

Like the term reduction T, the corresponding term graph reduction in Figure 6c is not weakly m-convergent: As we have illustrated in Example 5.1, the underlying sequence of term graphs is not convergent. On the other hand, the reduction does weakly p-converge to the term graph g_ω, which unravels to the term t to which the reduction T weakly p-converges to.

Similarly, also the reduction in Figure 6d follows its term rewriting counterpart T' closely: It both weakly m- and p-converges to the term graph h_ω, which unravels to the term t' that T' weakly m- and p-converges to. Example 5.1 and 4.1 explain how these limits come about.

Due to its higher degree of sharing, the rule ρ_2 permits to arrive at essentially the same result by a single reduction step as seen in Figure 6b. The resulting cyclic term graph h unravels to the same term t' as h_ω. The ρ_1-step that follows illustrates the interaction of rewrite rules with cycles. In fact, if we continue applying the rule ρ_1 after h', we obtain a reduction that weakly m- and p-converges to h_ω.

6.3. *m-convergence vs. p-convergence*

Recall that weak p-convergence in term rewriting is a conservative extension of weak m-convergence (cf. Theorem 2.1). The key property that makes this possible is that for each sequence $(t_\iota)_{\iota<\alpha}$ in $\mathcal{T}^\infty(\Sigma)$, we have that $\lim_{\iota\to\alpha} t_\iota = \liminf_{\iota\to\alpha} t_\iota$ whenever $(t_\iota)_{\iota<\alpha}$ converges, or $\liminf_{\iota\to\alpha} t_\iota$ is a total term.

Unfortunately, this is not the case for the metric space and the partial order that we consider on term graphs. As we have shown in Example 5.1, the sequence of term graphs depicted in Figure 3 has a total term graph as its limit inferior although it does not converge in the metric space. In fact, since the sequence in Figure 3 alternates between two distinct term graphs, it does not converge in any Hausdorff space, i.e. in particular, it does not converge in any metric space.

This example shows that we cannot hope to generalise the compatibility property that we have for terms: even if a sequence of total term graphs has a total term graph as its limit inferior, it might not converge. However, the other direction of the compatibility does hold true:

Theorem 6.1. If $(g_\iota)_{\iota<\alpha}$ converges, then $\lim_{\iota\to\alpha} g_\iota = \liminf_{\iota\to\alpha} g_\iota$.

Proof. In order to prove this property, we will use the construction of the limit and the limit inferior of a sequence of term graphs, which we have shown in Theorem 5.1 and Corollary 4.2, respectively.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

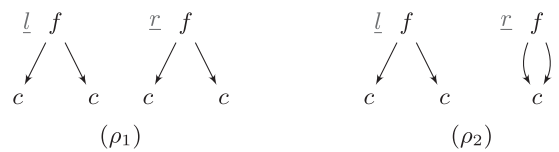


Fig. 7. Two term graph rules.

According to Theorem 5.1, we have that the canonical term graph $\lim_{\iota\to\alpha} g_\iota$ is given by the following labelled quotient tree $(P, l, \sim)$:

$$P = \bigcup_{\beta<\alpha} \bigcap_{\beta\leqslant\iota<\alpha} \mathcal{P}(g_\iota) \qquad \sim = \bigcup_{\beta<\alpha} \bigcap_{\beta\leqslant\iota<\alpha} \sim_{g_\iota}$$

$$l(\pi) = f \quad \text{iff} \quad \exists\beta < \alpha \forall \beta \leqslant \iota < \alpha : g_\iota(\pi) = f$$

We will show that $g = \liminf_{\iota\to\alpha} g_\iota$ induces the same labelled quotient tree.

From Corollary 4.2, we immediately obtain that $\mathcal{P}(g) \subseteq P$. To show the converse direction $\mathcal{P}(g) \supseteq P$, we assume some $\pi \in P$. According to Corollary 4.2, in order to show that $\pi \in \mathcal{P}(g)$, we have to find a $\beta < \alpha$ such that $\pi \in \mathcal{P}(g_\beta)$ and for each $\pi' < \pi$ there is some $f \in \Sigma_\perp$ such that $g_\iota(\pi') = f$ for all $\beta \leqslant \iota < \alpha$.

Because $\pi \in P$, there is some $\beta_1 < \alpha$ such that $\pi \in \mathcal{P}(g_\iota)$ for all $\beta_1 \leqslant \iota < \alpha$. Since $(g_\iota)_{\iota<\alpha}$ converges, it is also Cauchy. Hence, by Lemma 5.1, for each $d < \omega$, there is some $\beta_2 < \alpha$ such that $g_\gamma \dagger d \cong g_\iota \dagger d$ for all $\beta_2 \leqslant \gamma, \iota < \alpha$. By specialising this to $d = |\pi|$, we obtain some $\beta_2 < \alpha$ with $g_\gamma \dagger |\pi| \cong g_\iota \dagger |\pi|$ for all $\beta_2 \leqslant \gamma, \iota < \alpha$. Let $\beta = \max\{\beta_1, \beta_2\}$. Then we have $\pi \in \mathcal{P}(g_\iota)$ and $g_\beta \dagger |\pi| \cong g_\iota \dagger |\pi|$ for each $\beta \leqslant \iota < \alpha$. Hence, for each $\pi' < \pi$, the symbol $f = g_\beta(\pi')$ is well-defined, and, according to Corollary 5.1, we have that $g_\iota(\pi') = f$ for each $\beta \leqslant \iota < \alpha$.

The equalities $\sim = \sim_g$ and $l = g(\cdot)$ follow from Corollary 4.2 as $P = \mathcal{P}(g)$. □

From this property, we immediately obtain the following relation between weak *m*- and *p*-convergence:

Theorem 6.2. Let S be a reduction in a GRS $\mathcal{R}$.

$$\text{If} \quad S : g \overset{m}{\hookrightarrow}_{\mathcal{R}} h, \qquad \text{then} \qquad S : g \overset{p}{\hookrightarrow}_{\mathcal{R}} h.$$

Proof. Follows straightforwardly from Theorem 6.1. □

However, as we have indicated, weak *m*-convergence is not the total fragment of weak *p*-convergence as it is the case for TRSs. The GRS with the two rules depicted in Figure 7 yields the reduction sequence shown in Figure 3. This reduction weakly *p*-converges to $f(c, c)$ but is not weakly *m*-convergent.

7. Preservation of convergence through unravelling

In this section, we shall show that the convergence behaviour of term graph sequences – both in terms of metric limit and in terms of the limit inferior – is preserved by the

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

unravelling of term graphs to terms. As we will also show that the metric $\mathbf{d}_\dagger$ and partial order $\leqslant_\bot^{\mathsf{S}}$ coincide with the metric $\mathbf{d}$ and the partial order $\leqslant_\bot$ if restricted to terms, the preservation of convergence will show that both modes of convergence are sound w.r.t. the modes of convergence used in infinitary term rewriting.

The cornerstone of the investigation of unravellings is the following characterisation in terms of labelled quotient trees:

Proposition 7.1. The unravelling $\mathcal{U}(g)$ of a term graph $g \in \mathcal{G}^\infty(\Sigma)$ is given by the labelled quotient tree $(\mathcal{P}(g), g(\cdot), \mathcal{I}_{\mathcal{P}(g)})$.

Proof. Since $\mathcal{I}_{\mathcal{P}(g)}$ is a subrelation of $\sim_g$, we know that $(\mathcal{P}(g), g(\cdot), \mathcal{I}_{\mathcal{P}(g)})$ is a labelled quotient tree and thus uniquely determines a term tree t. By Lemma 3.3, there is a homomorphism from t to g. Hence, $\mathcal{U}(g) = t$. □

7.1. *Metric convergence*

We start with a specialisation of Lemma 5.2, which provides a characterisation of the simple truncation, to term trees:

Lemma 7.1. Let $t \in \mathcal{T}^\infty(\Sigma_\bot)$ and $d \leqslant \omega + 1$. The simple truncation $t\dagger d$ is given by the labelled quotient tree $(P, l, \mathcal{I}_P)$ with

$$P = \{\pi \in \mathcal{P}(t) \mid |\pi| \leqslant d\} \qquad l(\pi) = \begin{cases} t(\pi) & \text{if } |\pi| < d \\ \bot & \text{if } |\pi| \geqslant d \end{cases}$$

Proof. Immediate from Lemma 5.2 and the fact that $\sim_t$ is the identity relation $\mathcal{I}_{\mathcal{P}(t)}$ on $\mathcal{P}(t)$. □

This shows that the metric $\mathbf{d}_\dagger$ restricted to terms coincides with the metric $\mathbf{d}$ on terms. Moreover, we can use this in order to relate the metric distance between term graphs and the metric distance between their unravellings.

Lemma 7.2. For all $g, h \in \mathcal{G}^\infty(\Sigma)$, we have that $\mathbf{d}_\dagger(g, h) \geqslant \mathbf{d}_\dagger(\mathcal{U}(g), \mathcal{U}(h))$.

Proof. Let $d = \mathsf{sim}_\dagger(g, h)$. Hence, $g\dagger d \cong h\dagger d$ and we can assume that the corresponding labelled quotient trees as characterised by Lemma 5.2 coincide. We only need to show that $\mathcal{U}(g)\dagger d \cong \mathcal{U}(h)\dagger d$ since then $\mathsf{sim}_\dagger(\mathcal{U}(g), \mathcal{U}(h)) \geqslant d$ and thus $\mathbf{d}_\dagger(\mathcal{U}(g), \mathcal{U}(h)) \leqslant 2^{-d} = \mathbf{d}_\dagger(g, h)$. In order to show this, we show that the labelled quotient trees of $\mathcal{U}(g)\dagger d$ and $\mathcal{U}(h)\dagger d$ as characterised by Lemma 7.1 coincide. For the set of positions, we have the

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

following:

$$\begin{aligned}
&\pi \in \mathcal{P}(\mathcal{U}(g)\dagger d) \\
\iff &\pi \in \mathcal{P}(\mathcal{U}(g)), \quad |\pi| \leqslant d && \text{(Lemma 7.1)} \\
\iff &\pi \in \mathcal{P}(g), \quad |\pi| \leqslant d && \text{(Proposition 7.1)} \\
\iff &\pi \in \mathcal{P}(g\dagger d), \quad |\pi| \leqslant d && \text{(Corollary 5.1)} \\
\iff &\pi \in \mathcal{P}(h\dagger d), \quad |\pi| \leqslant d && (g\dagger d \cong h\dagger d) \\
\iff &\pi \in \mathcal{P}(h), \quad |\pi| \leqslant d && \text{(Corollary 5.1)} \\
\iff &\pi \in \mathcal{P}(\mathcal{U}(h)), \quad |\pi| \leqslant d && \text{(Proposition 7.1)} \\
\iff &\pi \in \mathcal{P}(\mathcal{U}(h)\dagger d) && \text{(Lemma 7.1)}
\end{aligned}$$

In order to show that the labellings are equal, consider some $\pi \in \mathcal{P}(\mathcal{U}(g)\dagger d)$ and assume at first that $|\pi| \geqslant d$. By Lemma 7.1, we then have $(\mathcal{U}(g)\dagger d)(\pi) = \bot = (\mathcal{U}(h)\dagger d)(\pi)$. Otherwise, if $|\pi| < d$, we obtain that

$$\begin{aligned}
(\mathcal{U}(g)\dagger d)(\pi) &\overset{\text{Lem. 7.1}}{=} \mathcal{U}(g)(\pi) \overset{\text{Prop. 7.1}}{=} g(\pi) \overset{\text{Cor. 5.1}}{=} g\dagger d(\pi) \\
&\overset{g\dagger d\cong h\dagger d}{=} h\dagger d(\pi) \overset{\text{Cor. 5.1}}{=} h(\pi) \overset{\text{Prop. 7.1}}{=} \mathcal{U}(h)(\pi) \overset{\text{Lem. 7.1}}{=} (\mathcal{U}(h)\dagger d)(\pi).
\end{aligned}$$ □

This immediately yields that Cauchy sequences are preserved by unravelling:

Lemma 7.3. If $(g_\iota)_{\iota<\alpha}$ is a Cauchy sequence in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$, then so is $(\mathcal{U}(g_\iota))_{\iota<\alpha}$.

Proof. This follows immediately from Lemma 7.2. □

Moreover, we obtain that limits in the metric space $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$ are preserved by unravelling.

Theorem 7.1. For every sequence $(g_\iota)_{\iota<\alpha}$ in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$, we have that $\lim_{\iota\to\alpha} g_\iota = g$ implies $\lim_{\iota\to\alpha} \mathcal{U}(g_\iota) = \mathcal{U}(g)$.

Proof. According to Theorem 5.1, we have that $\mathcal{P}(g) = \liminf_{\iota\to\alpha} \mathcal{P}(g_\iota)$, and that $g(\pi) = g_\beta(\pi)$ for some $\beta < \alpha$ with $g_\iota(\pi) = g_\beta(\pi)$ for all $\beta \leqslant \iota < \alpha$. By Proposition 7.1, we then obtain $\mathcal{P}(\mathcal{U}(g)) = \liminf_{\iota\to\alpha} \mathcal{P}(\mathcal{U}(g_\iota))$, and that $\mathcal{U}(g)(\pi) = \mathcal{U}(g_\beta)(\pi)$ for some $\beta < \alpha$ with $\mathcal{U}(g_\iota)(\pi) = \mathcal{U}(g_\beta)(\pi)$ for all $\beta \leqslant \iota < \alpha$. Since by Lemma 7.3, $(\mathcal{U}(g_\iota))_{\iota<\alpha}$ is Cauchy, we can apply Theorem 5.1 to obtain that $\lim_{\iota\to\alpha} \mathcal{U}(g_\iota) = \mathcal{U}(g)$. □

Since Lemma 7.1 confirms that the metric $\mathbf{d}_\dagger$ restricted to terms coincides with the metric $\mathbf{d}$ on terms, we have that convergence on term graphs simulates convergence on terms: if $(g_\iota)_{\iota<\alpha}$ converges to g in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\dagger)$, then $(\mathcal{U}(g_\iota))_{\iota<\alpha}$ converges to $\mathcal{U}(g)$ in $(\mathcal{T}^\infty(\Sigma), \mathbf{d})$.

7.2. *Partial order convergence*

At first, we derive a characterisation of the partial order $\leqslant^{\mathsf{S}}_\bot$ on terms by specialising Corollary 4.1:

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Lemma 7.4. Given two terms $s, t \in \mathcal{T}^\infty(\Sigma_\perp)$, we have $s \leqslant^{\mathsf{S}}_\perp t$ iff $s(\pi) = t(\pi)$ for all $\pi \in \mathcal{P}(s)$ with $g(\pi) \in \Sigma$.

Proof. Immediate from Corollary 4.1. □

This shows that the partial order $\leqslant^{\mathsf{S}}_\perp$ on term graphs generalises the partial order $\leqslant_\perp$ on terms, i.e. $\leqslant^{\mathsf{S}}_\perp$ restricted to $\mathcal{T}^\infty(\Sigma_\perp)$ coincides with $\leqslant_\perp$.

From the above finding, we easily obtain that the partial order $\leqslant^{\mathsf{S}}_\perp$ as well as its induced limits are preserved by unravelling:

Theorem 7.2. In the partially ordered set $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\perp), \leqslant^{\mathsf{S}}_\perp)$, the following holds:

(i) Given two term graphs g, h, we have that $g \leqslant^{\mathsf{S}}_\perp h$ implies $\mathcal{U}(g) \leqslant^{\mathsf{S}}_\perp \mathcal{U}(h)$.
(ii) For each directed set G, we have that $\mathcal{U}\left(\bigsqcup_{g \in G} g\right) = \bigsqcup_{g \in G} \mathcal{U}(g)$.
(iii) For each non-empty set G, we have that $\mathcal{U}\left(\bigsqcap_{g \in G} g\right) = \bigsqcap_{g \in G} \mathcal{U}(g)$.
(iv) For each sequence $(g_\iota)_{\iota<\alpha}$, we have that $\mathcal{U}(\liminf_{\iota\to\alpha} g_\iota) = \liminf_{\iota\to\alpha} \mathcal{U}(g_\iota)$.

Proof. (i) By Corollary 4.1, $g \leqslant^{\mathsf{S}}_\perp h$ implies that $g(\pi) = h(\pi)$ for all $\pi \in \mathcal{P}(g)$ with $g(\pi) \in \Sigma$. By Proposition 7.1, we then have $\mathcal{U}(g)(\pi) = \mathcal{U}(h)(\pi)$ for all $\pi \in \mathcal{P}(\mathcal{U}(g))$ with $\mathcal{U}(g)(\pi) \in \Sigma$ which, by Lemma 7.4, implies $\mathcal{U}(g) \leqslant^{\mathsf{S}}_\perp \mathcal{U}(h)$.

By a similar argument (ii) and (iii) follow from the characterisation of lubs and glbs in Theorem 4.1 and Proposition 4.2 by using Proposition 7.1.

(iv) Follows from (ii) and (iii). □

Since Lemma 7.4 shows that $\leqslant^{\mathsf{S}}_\perp$ and $\leqslant_\perp$ coincide on $\mathcal{T}^\infty(\Sigma_\perp)$, we thus obtain that the limit inferior on term graphs simulates the limit inferior on terms: If $\liminf_{\iota\to\alpha} g_\iota = g$ in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\perp), \leqslant^{\mathsf{S}}_\perp)$, then $\liminf_{\iota\to\alpha} \mathcal{U}(g_\iota) = \mathcal{U}(g)$ in $(\mathcal{T}^\infty(\Sigma_\perp), \leqslant_\perp)$.

8. Finite term graphs

In this section, we want to study the simple partial order $\leqslant^{\mathsf{S}}_\perp$ and the simple metric $\mathbf{d}_\dagger$ on finite term graphs. On terms, the partial order $\leqslant_\perp$ and the metric $\mathbf{d}$ allow us to reconstruct the set of (partial) terms from the set of finite (partial) terms via *ideal completion* and *metric completion*, respectively. In the following, we shall show that this generalises to the setting of canonical term graphs.

8.1. *Finitary properties*

Since term graphs are finitely branching, we know that, in each term graph, there are only a finite number of positions of a bounded length:

Lemma 8.1 (Bounded positions are finite). Let $g \in \mathcal{G}^\infty(\Sigma)$ and $d < \omega$. Then there are only finitely many positions of length at most d in g, i.e. the set $\{\pi \in \mathcal{P}(g) \mid |\pi| \leqslant d\}$ is finite.

Proof. Straightforward induction on d. □

From this, we can immediately conclude that the simple truncation of a term graph yields a finite term graph:

Proposition 8.1 (Simple truncations are finite). For each $g \in \mathcal{G}^\infty(\Sigma_\perp)$ and $d < \omega$, the simple truncation $g\dagger d$ is finite, i.e. $g\dagger d \in \mathcal{G}(\Sigma_\perp)$.

Proof. By Lemma 8.1, the set $P = \{\pi \in \mathcal{P}(g) \mid |\pi| \leqslant d\}$ is finite. Since the function $f\colon P \to N^{g\dagger d}$ defined by $f(\pi) = \mathsf{node}_g(\pi)$ is surjective, we can conclude that $N^{g\dagger d}$ is finite. □

We know that positions describe the structure of a term graph. However, cycles cause infinite repetition of essentially the same structure of a position. Therefore, a finite term graph may have infinitely many positions. In the following, we want to avoid this by considering only *essential positions*:

Definition 8.1 (Redundant/essential positions). A position $\pi \in \mathcal{P}(g)$ in a term graph $g \in \mathcal{G}^\infty(\Sigma)$ is called *redundant* if there are $\pi_1, \pi_2 \in \mathcal{P}(g)$ with $\pi_1 < \pi_2 < \pi$ such that $\pi_1 \sim_g \pi_2$. A position that is not redundant is called *essential*. The set of all essential positions of g are denoted $\mathcal{P}^e(g)$; the set of all essential positions of a node n in g are denoted $\mathcal{P}^e_g(n)$.

Note that a position is redundant iff one of its proper prefixes is cyclic. This means that the set $\mathcal{P}^e(g)$ of essential positions is closed under prefixes.

Lemma 8.2 (Decomposition of redundant positions). For each $g \in \mathcal{G}^\infty(\Sigma)$ and $\pi \in \mathcal{P}(g)$, we have that π is redundant iff there are $\pi_1, \pi_2 \in \mathcal{P}^e(g)$ such that $\pi_1 < \pi_2 < \pi$ and $\pi_1 \sim_g \pi_2$.

Proof. The ‘if’ direction follows immediately from the definition of redundancy. We will show the ‘only if’ direction by induction on the length of π.

If π is redundant in g, then there are $\pi_1, \pi_2 \in \mathcal{P}(g)$ with $\pi_1 < \pi_2 < \pi$ and $\pi_1 \sim_g \pi_2$. If π_2 is essential, then also π_1 is essential since it is a prefix of π_2. Otherwise, if π_2 is redundant, we can apply the induction hypothesis to π_2 to obtain $\pi'_1, \pi'_2 \in \mathcal{P}^e(g)$ with $\pi'_1 < \pi'_2 < \pi_2$ and $\pi'_1 \sim_g \pi'_2$. □

With essential positions, we have a finite representation of the structure of term graphs even if the term graph is cyclic.

Proposition 8.2 (Essential positions characterise finiteness). A term graph $g \in \mathcal{G}^\infty(\Sigma)$ is finite iff $\mathcal{P}^e(g)$ is finite.

Proof. If g is finite, then let $n = |N^g|$. Whenever a position $\pi \in \mathcal{P}(g)$ is longer than n, then a proper prefix of π passes more than n nodes. By the pigeon hole principle, we thus know that there is a node that a proper prefix of π passes twice. Hence, π is redundant. Therefore, we know that every essential position must be of length at most n. Since, according to Lemma 8.1, there are only finitely many such positions in g, we know that $\mathcal{P}^e(g)$ is finite.

If g is infinite, we can apply König's Lemma to obtain an infinite acyclic path (starting in the root of g) that does not pass a node twice. Since each finite prefix of this path is an essential position, there are infinitely many essential positions. □

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Indeed, the essential positions of a term graph are sufficient in order to characterise the structure of term graphs in the form of Δ-homomorphisms:

Proposition 8.3 (Essential positions characterise Δ-homomorphisms). Given $g, h \in \mathcal{G}^\infty(\Sigma)$, there is a Δ-homomorphism $\phi\colon g \to_\Delta h$ iff, for all $\pi, \pi' \in \mathcal{P}^e(g)$, we have

(a) $\pi \sim_g \pi' \quad\Longrightarrow\quad \pi \sim_h \pi'$, and
(b) $g(\pi) = h(\pi) \quad$ whenever $\quad g(\pi) \notin \Delta$.

Proof. The 'only if' direction follows immediately from Lemma 3.3. For the converse direction, assume that both (a) and (b) hold. Define the function $\phi\colon N^g \to N^h$ by $\phi(n) = m$ iff $\mathcal{P}_g(n) \subseteq \mathcal{P}_h(m)$ for all $n \in N^g$ and $m \in N^h$. To confirm that this is well-defined, we show at first that, for each $n \in N^g$, there is at most one $m \in N^h$ with $\mathcal{P}_g(n) \subseteq \mathcal{P}_h(m)$. Suppose there is another node $m' \in N^h$ with $\mathcal{P}_g(n) \subseteq \mathcal{P}_h(m')$. Since $\mathcal{P}_g(n) \neq \emptyset$, this implies $\mathcal{P}_h(m) \cap \mathcal{P}_h(m') \neq \emptyset$. Hence, $m = m'$. Second, we show that there is at least one such node m. We know that each node has at least one essential position. Choose some $\pi^* \in \mathcal{P}^e_g(n)$. Since then $\pi^* \sim_g \pi^*$ and, by (a), also $\pi^* \sim_h \pi^*$ holds, there is some $m \in N^h$ with $\pi^* \in \mathcal{P}_h(m)$. Next, we show by induction on the length of π that $\pi \in \mathcal{P}_g(n)$ implies $\pi \in \mathcal{P}_h(m)$. If $\pi \in \mathcal{P}_g(n)$, then $\pi \sim_g \pi^*$. In case that π is essential in g, we obtain $\pi \sim_h \pi^*$ from (a) and thus $\pi \in \mathcal{P}_h(m)$. Otherwise, i.e. if π is redundant in g, we can decompose π into $\pi = \pi_1 \cdot \pi_2 \cdot \pi_3$ such that π_2 and π_3 are non-empty and $\pi_1 \sim_g \pi_1 \cdot \pi_2$. By Lemma 8.2, we can assume that π_1 and $\pi_1 \cdot \pi_2$ are essential in g. Hence, $\pi_1 \sim_g \pi_1 \cdot \pi_2$ implies, by (a), that $\pi_1 \sim_h \pi_1 \cdot \pi_2$. Moreover, $\pi_1 \sim_g \pi_1 \cdot \pi_2$ means that the prefix $\pi_1 \cdot \pi_2$ of π can be replaced by π_1 in g, i.e. $\pi_1 \cdot \pi_3 \in \mathcal{P}_g(n)$. Since $\pi_1 \cdot \pi_3$ is strictly shorter than π, we can apply the induction hypothesis to obtain that $\pi_1 \cdot \pi_3 \in \mathcal{P}_h(m)$. From this and from $\pi_1 \sim_h \pi_1 \cdot \pi_2$, we can then conclude that $\pi_1 \cdot \pi_2 \cdot \pi_3 \in \mathcal{P}_h(m)$.

Using Lemma 3.1, we can see that ϕ is a Δ-homomorphism from g to h: condition (a) of Lemma 3.1 follows immediately from the construction of ϕ and condition (b) of Lemma 3.1 follows from (b) since each node has at least one essential position. $\square$

Consequently, we immediately obtain a characterisation of the simple partial order $\leqslant^{\mathsf{S}}_\perp$ in terms of essential positions:

Corollary 8.1 (Essential positions characterise $\leqslant^{\mathsf{S}}_\perp$). Let $g, h \in \mathcal{G}^\infty(\Sigma_\perp)$. Then, $g \leqslant^{\mathsf{S}}_\perp h$ iff the following conditions are met:

(a) $\pi \sim_g \pi' \quad\Longrightarrow\quad \pi \sim_h \pi' \quad$ for all $\pi, \pi' \in \mathcal{P}^e(g)$.
(b) $g(\pi) = h(\pi) \quad$ for all $\pi \in \mathcal{P}^e(g)$ with $g(\pi) \in \Sigma$.

The above characterisation allows us to prove that the lub of a finite number of finite term graphs can only be finite as well:

Proposition 8.4 (lub of finite term graphs is finite). For each finite set $G \subseteq_{fin} \mathcal{G}_\mathcal{C}(\Sigma_\perp)$ with an upper bound in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\perp), \leqslant^{\mathsf{S}}_\perp)$, we have $\bigsqcup G \in \mathcal{G}_\mathcal{C}(\Sigma_\perp)$.

Proof. Let $G \subseteq_{fin} \mathcal{G}_\mathcal{C}(\Sigma_\perp)$ be a finite set with upper bound $\widehat{g}$. If G is empty, then $\bigsqcup G = \perp \in \mathcal{G}_\mathcal{C}(\Sigma_\perp)$. Otherwise, we know, by Proposition 2.1, that the complete semilattice $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\perp), \leqslant^{\mathsf{S}}_\perp)$ is also bounded complete. Hence, G has a least upper bound $\overline{g}$. Since $\overline{g}$

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

is an upper bound of G, we find for each $g \in G$ a $\bot$-homomorphism $\phi_g : g \to_\bot \overline{g}$. Let $N = \bigcup_{g \in G} Im(\phi_g)$ be the combined image of those $\bot$-homomorphisms. Since each $g \in G$ is finite, also their image $Im(\phi_g)$ is finite and thus so is N. We conclude the proof by showing that $N^{\overline{g}} \subseteq N$, which proves that $\overline{g}$ is finite.

We show that $n \in N^{\overline{g}}$ implies $n \in N$ by induction on $\mathsf{depth}_{\overline{g}}(n)$. If $\mathsf{depth}_{\overline{g}}(n) = 0$, then $n = r^{\overline{g}}$. Choose some $g \in G$. Since then $\phi_g(r^g) = r^{\overline{g}}$, we have that $n \in Im(\phi_g) \subseteq N$. If $\mathsf{depth}_{\overline{g}}(n) > 0$, then there is some $m \in N^{\overline{g}}$ with $\mathsf{depth}_{\overline{g}}(m) < \mathsf{depth}_{\overline{g}}(n)$ and $\mathsf{suc}_i^{\overline{g}}(m) = n$ for some i. Hence, we can apply the induction hypothesis which yields that $m \in N$. Since m has a successor in $\overline{g}$, we have that $\mathsf{lab}^{\overline{g}}(m) \in \Sigma$. Construct the term graph $\widehat{g}$ from $\overline{g}$ by relabelling m to $\bot$ and removing all its outgoing edges as well as all nodes that thus become unreachable. The mapping $\phi \colon N^{\widehat{g}} \to N^{\overline{g}}$ given by $\phi(\widehat{n}) = \widehat{n}$ for all $\widehat{n} \in N^{\widehat{g}}$ is a $\bot$-homomorphism. Thus, $\mathcal{C}(\widehat{g}) <_\bot^{\mathsf{S}} \overline{g}$. However, since $\overline{g}$ is the least upper bound of G, $\mathcal{C}(\widehat{g})$ cannot be an upper bound of G. But, for each $g \in G$, the mapping ϕ_g is also a $\bot$-homomorphism from g to $\widehat{g}$ provided each $m' \in N^g$ with $\phi_g(m') = m$ is labelled $\bot$ in g. Since this cannot be the case for all $g \in G$, we find some $g \in G, m' \in N^g$ such that $\phi_g(m') = m$ and $\mathsf{lab}^g(m') \in \Sigma$. Since ϕ_g is then homomorphic in m', we know that m' has an ith successor in g such that

$$\phi_g(\mathsf{suc}_i^g(m')) = \mathsf{suc}_i^{\overline{g}}(\phi_g(m')) = \mathsf{suc}_i^{\overline{g}}(m) = n.$$

Hence, $n \in Im(\phi_g) \subseteq N$. □

8.2. *Ideal completion*

In this section, we shall show that the set $\mathcal{G}_\mathcal{C}^\infty(\Sigma_\bot)$ of (potentially infinite) canonical term graphs can be constructed from the set $\mathcal{G}_\mathcal{C}(\Sigma_\bot)$ of finite canonical term graphs via the ideal completion of the partially ordered set $(\mathcal{G}_\mathcal{C}(\Sigma_\bot), \leqslant_\bot^{\mathsf{S}})$.

Given a partially order set, its ideal completion provides an extension of the original partially ordered set that is a cpo.

Definition 8.2 (Ideal, ideal completion). Let $(A, \leqslant)$ be a partially ordered set and $B \subseteq A$.

(i) The set B is called *downward-closed* if for all $a \in A, b \in B$ with $a \leqslant b$, we have that $a \in B$.

(ii) The set B is called an *ideal* if it is directed and downward-closed. We write $\mathsf{Idl}(A, \leqslant)$ to denoted the set of all ideals of $(A, \leqslant)$.

(iii) The *ideal completion* of $(A, \leqslant)$, is the partially ordered set $(\mathsf{Idl}(A, \leqslant), \subseteq)$.

For terms, we already know that the set of (potentially infinite) terms can be constructed by forming the ideal completion of the partially ordered set $(\mathcal{T}(\Sigma_\bot), \leqslant_\bot)$ of finite terms.

Theorem 8.1 (Ideal completion of terms, Berry and Lévy (1977)). The ideal completion of $(\mathcal{T}(\Sigma_\bot), \leqslant_\bot)$ is order isomorphic to $(\mathcal{T}^\infty(\Sigma_\bot), \leqslant_\bot)$.

We show an analogous result for term graphs:

Theorem 8.2 (Ideal completion of term graphs). The ideal completion of the partially ordered set $(\mathcal{G}_\mathcal{C}(\Sigma_\bot), \leqslant_\bot^{\mathsf{S}})$ is order isomorphic to $(\mathcal{G}_\mathcal{C}^\infty(\Sigma_\bot), \leqslant_\bot^{\mathsf{S}})$.

Proof. Let I be the set $\mathsf{Idl}(\mathcal{G}_\mathcal{C}(\Sigma_\bot), \leqslant^\mathsf{S}_\bot)$ of ideals in $(\mathcal{G}_\mathcal{C}(\Sigma_\bot), \leqslant^\mathsf{S}_\bot)$. To prove that $(I, \subseteq)$ and $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot), \leqslant^\mathsf{S}_\bot)$ are order isomorphic, we will construct two monotonic functions $\phi\colon \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot) \to I$ and $\psi\colon I \to \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$, and show that they are inverses of each other.

Define the function ϕ as follows: $\phi(g) = \left\{h \in \mathcal{G}_\mathcal{C}(\Sigma_\bot) \,\middle|\, h \leqslant^\mathsf{S}_\bot g\right\}$ for all $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$. We have to show that $\phi(g)$ is indeed an ideal for each $g \in \mathcal{G}_\mathcal{C}(\Sigma_\bot)$. By definition, $\phi(g)$ is downward-closed. To show that it is directed, let $h_1, h_2 \in \phi(g)$, i.e. $h_1, h_2 \leqslant^\mathsf{S}_\bot g$. By Proposition 8.4, $\{h_1, h_2\}$ has a least upper bound h in $\mathcal{G}_\mathcal{C}(\Sigma_\bot)$. Since g is an upper bound of $\{h_1, h_2\}$, we have $h \leqslant^\mathsf{S}_\bot g$ and thus $h \in \phi(g)$.

Monotonicity of ϕ follows immediately from its definition.

Define the function ψ as follows: $\psi(G) = \bigsqcup G$ for all $G \in I$. Since according to Theorem 4.1, $(\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot), \leqslant^\mathsf{S}_\bot)$ is a cpo, we know that ψ is well-defined. The monotonicity of ψ follows immediately from its definition.

Finally, we show that ϕ and ψ are inverses of each other. At first, we show that $\psi(\phi(g)) = g$ for all $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$, i.e. $g = \bigsqcup \phi(g)$. By definition of ϕ, we already know that g is an upper bound of $\phi(g)$. To show that it is the least upper bound, we assume that $\overline{g} \in \mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$ is an upper bound of $\phi(g)$ and show that $g \leqslant^\mathsf{S}_\bot \overline{g}$. We will do that by using Corollary 4.1.

(a) Let $\pi_1 \sim_g \pi_2$ and let $d = \max\{|\pi_1|, |\pi_2|\}$. Then, according to Corollary 5.1, also $\pi_1 \sim_{g\dagger d} \pi_2$. Moreover, by Proposition 8.1, $g\dagger d$ is finite and, by Corollary 5.2, $g\dagger d \leqslant^\mathsf{S}_\bot g$. Hence, since $g\dagger d \in \phi(g)$ and thus $g\dagger d \leqslant^\mathsf{S}_\bot \overline{g}$. This means that $\pi_1 \sim_{g\dagger d} \pi_2$ implies $\pi_1 \sim_{\overline{g}} \pi_2$, according to Corollary 4.1.

(b) Let $g(\pi) = f \in \Sigma$ and let $d = 1 + |\pi|$. Then, according to Corollary 5.1, also $g\dagger d(\pi) = f$. As for (a), we know that $g\dagger d \leqslant^\mathsf{S}_\bot \overline{g}$, which implies $\overline{g}(\pi) = f$, by Corollary 4.1.

Last, we show that $\phi(\psi(G)) = G$ for all $G \in I$. The inclusion $\phi(\psi(G)) \supseteq G$ is easy to prove: If $g \in G$, then $g \leqslant^\mathsf{S}_\bot \bigsqcup G$, and therefore $g \in \phi(\psi(G))$. For the converse inclusion, assume that $h \in \phi(\psi(G))$, i.e. $h \in \mathcal{G}_\mathcal{C}(\Sigma_\bot)$ with $h \leqslant^\mathsf{S}_\bot \bigsqcup G$. We claim that there is some $\widehat{h} \in G$ with $h \leqslant^\mathsf{S}_\bot \widehat{h}$. Since G is downward-closed, this then implies $h \in G$. We conclude this proof by constructing a $\widehat{h} \in G$ with $h \leqslant^\mathsf{S}_\bot \widehat{h}$.

Let $\overline{g} = \bigsqcup G$. Since $h \leqslant^\mathsf{S}_\bot \overline{g}$, we have by Corollary 8.1 that $\pi \sim_h \pi'$ implies $\pi \sim_{\overline{g}} \pi'$ for all $\pi, \pi' \in \mathcal{P}^e(h)$. In turn, $\pi \sim_{\overline{g}} \pi'$ implies by Theorem 4.1, that there is some $g \in G$ with $\pi \sim_g \pi'$. According to Proposition 8.2, the set $\mathcal{P}^e(h)$ is finite and thus there are only finitely many pairs $\pi, \pi' \in \mathcal{P}^e(h)$. Hence, we find a finite set $H \subseteq G$ such that for each $\pi, \pi' \in \mathcal{P}^e(h)$ with $\pi \sim_h \pi'$, there is a $g \in H$ with $\pi \sim_g \pi'$. Since H is a finite subset of the directed set G, there is some $h_1 \in G$ that is an upper bound of H. Consequently, for each $\pi, \pi' \in \mathcal{P}^e(h)$ with $\pi \sim_h \pi'$, we have $\pi \sim_{h_1} \pi'$ by Corollary 8.1.

By a similar argument, we find some $h_2 \in G$ such that for each $\pi \in \mathcal{P}^e(h)$ with $h(\pi) = f \in \Sigma$, we have $h_2(\pi) = f$. Since G is directed, we find some $\widehat{h} \in G$ with $h_1, h_2 \leqslant^\mathsf{S}_\bot \widehat{h}$. Hence, by Corollary 8.1, for all $\pi, \pi' \in \mathcal{P}^e(h)$, we have that $\pi \sim_h \pi'$ implies $\pi \sim_{\widehat{h}} \pi'$ and that $h(\pi) = f \in \Sigma$ implies $\widehat{h}(\pi) = f$. According to Corollary 8.1, this means that $h \leqslant^\mathsf{S}_\bot \widehat{h}$. □

The above theorem show a certain completeness of the partial order $\leqslant^\mathsf{S}_\bot$ in the sense that it allows us to canonically construct the set of term graphs $\mathcal{G}^\infty_\mathcal{C}(\Sigma_\bot)$ from the

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

set of finite term graphs $\mathcal{G}_{\mathcal{C}}(\Sigma_\perp)$. More concretely, an infinite term graph $g \in \mathcal{G}^\infty_{\mathcal{C}}(\Sigma_\perp)$ can be constructed by a limit construction involving only finite term graphs, viz. $g = \bigsqcup \left\{ h \in \mathcal{G}_{\mathcal{C}}(\Sigma_\perp) \,\middle|\, h \leqslant^{\mathsf{S}}_\perp g \right\}$. In fact, such a construction can also be achieved by the limit inferior of a sequence of finite graphs since we have that $g = \liminf_{d \to \omega} g\dagger d$.

Such a representation of infinite term graphs as a lub or a limit inferior of a sequence of finite term graphs is not possible for the rigid partial order $\leqslant^{\mathsf{R}}_\perp$. For example, there is no set of finite term graphs G whose lub is the term graph h_ω from Figure 6d w.r.t. the partial order $\leqslant^{\mathsf{R}}_\perp$. The reason is that no finite term graph g with $g \leqslant^{\mathsf{R}}_\perp h_\omega$ has a node labelled b at position $\langle 0 \rangle$.

8.3. *Metric completion*

In this section, we shall show that the set $\mathcal{G}^\infty_{\mathcal{C}}(\Sigma)$ of (potentially infinite) canonical term graphs can also be obtained as the metric completion of the metric space $(\mathcal{G}_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$ of finite term graphs endowed with the simple metric $\mathbf{d}_\dagger$.

Analogous to the ideal completion of partially ordered sets, the metric completion extends a metric spaces to a complete metric space.

Definition 8.3. Let $(M, \mathbf{d})$ be a metric space. The *closure* of a subset $N \subseteq M$, denoted $\mathcal{C}l\,(N)$, is the set $\{x \in M \mid x \text{ is the limit of a sequence in } N\}$. A subset $N \subseteq M$ is called dense if $\mathcal{C}l\,(N) = M$. A complete metric space $(M^\bullet, \mathbf{d}^\bullet)$ is called the *metric completion* of $(M, \mathbf{d})$ if there is an isometric embedding ϕ from $(M, \mathbf{d})$ into $(M^\bullet, \mathbf{d}^\bullet)$ and if the image $Im(\phi)$ of ϕ is dense in $(M^\bullet, \mathbf{d}^\bullet)$.

The metric completion of a metric space is unique up to isometry.

Again, for terms, we already know that we can construct the set of (potentially infinite) terms $\mathcal{T}^\infty(\Sigma)$ as the metric completion of the metric space $(\mathcal{T}(\Sigma), \mathbf{d})$ of finite terms.

Theorem 8.3 (Metric completion of terms, Barr (1993)). The metric completion of $(\mathcal{T}(\Sigma), \mathbf{d})$ is the metric space $(\mathcal{T}^\infty(\Sigma), \mathbf{d})$.

Analogously, we can show that the metric space $(\mathcal{G}^\infty_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$ of (potentially infinite) term graphs arises as the metric completion of the metric space $(\mathcal{G}_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$ of finite term graphs.

Theorem 8.4 (Metric completion of term graphs). The metric completion of $(\mathcal{G}_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$ is the metric space $(\mathcal{G}^\infty_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$.

Proof. Since $\mathcal{G}_{\mathcal{C}}(\Sigma)$ is a subset of $\mathcal{G}^\infty_{\mathcal{C}}(\Sigma)$, we can define the isometric embedding $\phi\colon \mathcal{G}_{\mathcal{C}}(\Sigma) \to \mathcal{G}^\infty_{\mathcal{C}}(\Sigma)$ by setting $\phi(g) = g$. It only remains to be shown that $Im(\phi) = \mathcal{G}_{\mathcal{C}}(\Sigma)$ is dense in $(\mathcal{G}^\infty_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$. This is achieved by showing that for each $g \in \mathcal{G}^\infty_{\mathcal{C}}(\Sigma)$, we find a sequence $(g_i)_{i<\omega}$ in $\mathcal{G}_{\mathcal{C}}(\Sigma)$ that converges to g. From its definition, it is clear that the simple truncation is idempotent, i.e. $(g\dagger d)\dagger d = g\dagger d$, for all $d < \omega$. Hence, by Lemma 5.1, the sequence $(g\dagger d)_{d<\omega}$ converges to g in $(\mathcal{G}^\infty_{\mathcal{C}}(\Sigma), \mathbf{d}_\dagger)$. Moreover, according to Proposition 8.1, $(g\dagger d)_{d<\omega}$ is a sequence in $\mathcal{G}_{\mathcal{C}}(\Sigma)$. □

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

The above theorem shows that the metric $\mathbf{d}_\dagger$ is complete in the sense that it allows us to construct the set of term graphs $\mathcal{G}^\infty_\mathcal{C}(\Sigma)$ from the set of finite term graphs $\mathcal{G}_\mathcal{C}(\Sigma)$ in a canonical way. More concretely, each term graph $g \in \mathcal{G}^\infty_\mathcal{C}(\Sigma)$ can be constructed as the limit of a sequence of finite term graphs, viz. $g = \lim_{d\to\omega} g\dagger d$.

We cannot obtain such a completeness result for the rigid metric $\mathbf{d}_\ddagger$. For instance, consider the term graph h_ω from Figure 6d. For each $d > 1$, the rigid truncation $h_\omega\ddagger d$ of h_ω is equal to h_ω itself. Hence, there is no finite term graph g with a similarity $\mathsf{sim}_\ddagger(g, h_\omega) > 1$, which means, according to Lemma 5.1, that there is no sequence of finite term graphs that converges to h_ω in $(\mathcal{G}^\infty_\mathcal{C}(\Sigma), \mathbf{d}_\ddagger)$.

9. Concluding remarks

We have devised two independently defined but closely related infinitary calculi of term graph rewriting. Whilst this is not the first proposal for infinitary term graph rewriting calculi, we gave several arguments why the present approach is superior to our previous approach (Bahr 2012b): It is more natural, simpler and less restrictive. Due to the findings we have obtained here, we are very confident that we found two appropriate notions of convergence that generalise the corresponding notions of convergence on terms.

There is, however, one aspect of our notion of convergence that might be interpreted as an argument against its appropriateness. On term graphs, we do not obtain the correspondence between *p*- and *m*-convergence known from infinitary term rewriting; cf. Theorem 2.1. The underlying reason for the discrepancy is the fact that the partial order on term graphs $\leqslant^{\mathsf{S}}_\bot$ does not only capture the level of partiality – like $\leqslant_\bot$ does on terms – but also the degree of sharing. However, this discrepancy might just be a manifestation of the fundamental difference between terms and term graphs – namely sharing.

Unfortunately, we do not have solid soundness or completeness results apart from the preservation of convergence under unravelling and the metric/ideal completion construction of the set of term graphs. Even establishing soundness turns out to be difficult in the setting of weak convergence.

If we shift from the weak notions of convergence studied here to strong notions of convergence – in analogy to strong convergence in infinitary term rewriting (Kennaway et al. 1995) – all of the above-mentioned shortcomings disappear (Bahr 2012a). In particular, we regain the correspondence between metric and partial order convergence: Strong *p*-convergence on term graphs is a conservative extension of strong *m*-convergence. Moreover, with the move to strong convergence, it is also possible to establish that infinitary term graph rewriting is sound and complete w.r.t. term rewriting.

These additional findings further substantiate our claim that the fundamental structures that we have studied here are appropriate generalisations of the corresponding structures on terms for the formalisation of convergence in rewriting.

Acknowledgement

The author wishes to thank Clemens Grabmayer for his remarks on an earlier version of this paper.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

References

Ariola, Z. and Blom, S. (2005). Skew and ω-skew confluence and abstract Böhm semantics. Middeldorp, A., van Oostrom, V., van Raamsdonk, F. and de Vrijer, R. (eds.) *Processes, Terms and Cycles: Steps on the Road to Infinity*, Lecture Notes in Computer Science, vol. 3838, Springer, Berlin, Heidelberg, 368–403.

Ariola, Z. M. and Blom, S. (2002). Skew confluence and the lambda calculus with letrec. *Annals of Pure and Applied Logic* **117** (1–3) 95–168.

Ariola, Z.M. and Klop, J.W. (1997). Lambda calculus with explicit recursion. *Information and Computation* **139** (2) 154–233.

Arnold, A. and Nivat, M. (1980). The metric space of infinite trees. Algebraic and topological properties. *Fundamenta Informaticae* **3** (4) 445–476.

Bahr, P. (2009). *Infinitary Rewriting – Theory and Applications*. Master's Thesis, Vienna University of Technology, Vienna.

Bahr, P. (2010). Abstract models of transfinite reductions. In Lynch, C. (ed.) *Proceedings of the 21st International Conference on Rewriting Techniques and Applications*, Leibniz International Proceedings in Informatics, vol. 6, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 49–66.

Bahr, P. (2012a). Infinitary term graph rewriting is simple, sound and complete. Tiwari, A. (ed.) *Proceedings of the 23rd International Conference on Rewriting Techniques and Applications*, Leibniz International Proceedings in Informatics, vol. 15, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 69–84.

Bahr, P. (2012b). Modes of convergence for term graph rewriting. *Logical Methods in Computer Science* **8** (2) 1–60.

Bahr, P. (2014). Partial order infinitary term rewriting. *Logical Methods in Computer Science* **10** (2) 1–52.

Barendregt, H.P., van Eekelen, M.C.J.D., Glauert, J.R.W., Kennaway, R., Plasmeijer, M.J. and Sleep, M.R. (1987). Term graph rewriting. In: de Bakker A. J. Nijman and Treleaven, P.C. (ed.) *Parallel Architectures and Languages Europe, Volume II: Parallel Languages*. Lecture Notes in Computer Science, vol. 259, Springer, Berlin, Heidelberg, 141–158.

Barendsen, E. (2003). Term graph rewriting. In: Terese (ed.) *Term Rewriting Systems*, 1st edn., Cambridge University Press, Chap. 13, 712–743.

Barr, M. (1993). Terminal coalgebras in well-founded set theory. *Theoretical Computer Science* **114** (2) 299–315.

Berry, G. and Lévy, J.-J. (1977). Minimal and optimal computations of recursive programs. In: *Proceedings of the 4th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages*. ACM, New York, NY, USA, 215–226.

Dershowitz, N., Kaplan, S. and Plaisted, D.A. (1991). Rewrite, rewrite, rewrite, rewrite, rewrite, ... *Theoretical Computer Science* **83** (1) 71–96.

Goguen, J.A., Thatcher, J.W., Wagner, E.G. and Wright, J.B. (1977). Initial algebra semantics and continuous algebras. *Journal of the ACM* **24** (1) 68–95.

Henderson, P. and Morris Jr., J.H. (1976). A lazy evaluator. In: *Proceedings of the 3rd ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages*. ACM, New York, NY, USA, 95–103.

Hughes, J. (1989). Why functional programming matters. *The Computer Journal* **32** (2) 98–107.

Kahn, G. and Plotkin, G.D. (1993). Concrete domains. *Theoretical Computer Science* **121** (1–2) 187–277.

Kelley, J.L. (1955). *General Topology*, Graduate Texts in Mathematics, vol. 27, Springer-Verlag.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press

Kennaway, R. (1992). On transfinite abstract reduction systems. Technical report, CWI (Centre for Mathematics and Computer Science), Amsterdam.

Kennaway, R. and de Vries, F.-J. (2003). Infinitary rewriting. In: Terese (ed.) *Term Rewriting Systems*, 1st edn., Cambridge University Press, Chap. 12, 668–711.

Kennaway, R., Klop, J. W., Sleep, M.R. and de Vries, F.-J. (1994). On the adequacy of graph rewriting for simulating term rewriting. *ACM Transactions on Programming Languages and Systems* **16** (3) 493–523.

Kennaway, R., Klop, J.W., Sleep, M.R. and de Vries, F.-J. (1995). Transfinite reductions in orthogonal term rewriting systems. *Information and Computation* **119** (1) 18–38.

Marlow, S. (2010). *Haskell 2010 Language Report*. Available at: https://www.haskell.org/onlinereport/haskell2010/

Peyton-Jones, S. (1987). *The Implementation of Functional Programming Languages*, Prentice Hall.

Plasmeijer, R. and van Eekelen, M.C.J.D. (1993). *Functional Programming and Parallel Graph Rewriting*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Plump, D. (1999). Term graph rewriting. In: Ehrig, H., Engels, G., Kreowski, H.-J. and Rozenberg, G. (eds.) *Handbook of Graph Grammars and Computing by Graph Transformation, Volume 2: Applications, Languages, and Tools*, World Scientific Publishing Co., Inc. River Edge, NJ, USA, 3–61.

Terese (2003). *Term Rewriting Systems*, 1st edn., Cambridge University Press.

https://doi.org/10.1017/S0960129518000166 Published online by Cambridge University Press