C e n t r u m   v o o r   W i s k u n d e   e n   I n f o r m a t i c a

REPORT_RAPPORT_

SEN

Software Engineering

Software ENgineering

Lifting infinite normal form definitions from term rewriting to term graph rewriting

S.C.C. Blom

REPORT SEN-R0225 DECEMBER 31, 2002

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-369X

Lifting Infinite Normal Form Definitions from Term Rewriting to Term Graph Rewriting

Stefan Blom
email: sccblom@cwi.nl

CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

ABSTRACT
Infinite normal forms are a way of giving semantics to non-terminating rewrite systems. The notion is a generalization of the Böhm tree in the lambda calculus. It was first introduced in [AB97] to provide semantics for a lambda calculus on terms with letrec. In that paper infinite normal forms were defined directly on the graph rewrite system. In [Blo01] the framework was improved by defining the infinite normal form of a term graph using the infinite normal form on terms. This approach of lifting the definition makes the non-confluence problems introduced into term graph rewriting by substitution rules much easier to deal with. In this paper, we give a simplified presentation of the latter approach.

2000 Mathematics Subject Classification: 68Q42, 68Q55
Keywords and Phrases: term graph, Böhm tree, infinite term, rewriting, lambda calculus, combinatory reduction system
Note: Work carried out under project SEN2 "Specification and Analysis of Embedded Systems"

1. Introduction

A simple way of representing term graphs is to use the letrec syntax. By using this syntax, we can derive term graph rewrite systems from term rewrite systems. For example, from the β-rule in the lambda calculus

$$(\lambda x.M)\,N \xrightarrow[\beta]{} M[x:=N] \ ,$$

we can derive

$$\begin{array}{lcl} (\lambda x.M)\,N & \xrightarrow[\beta\circ]{} & \texttt{letrec}\,x=N\,\texttt{in}\,M \ ; \\ \texttt{letrec}\,x=M,D\,\texttt{in}\,C[x] & \xrightarrow[\text{es}]{} & \texttt{letrec}\,x=M,D\,\texttt{in}\,C[M] \ ; \\ (\texttt{letrec}\,D\,\texttt{in}\,M)\,N & \xrightarrow[\text{lift}]{} & \texttt{letrec}\,D\,\texttt{in}\,(M\,N) \ . \end{array} \tag{1.1}$$

(See [AK97] for details on exactly how this rewrite system may be derived.) We will refer to these derived systems as cyclic extensions.

The Böhm tree and the Lévy-Longo tree are nice notions of semantics for the lambda calculus. It is easy to extend the theory of these trees to the simple cyclic extension given above. In [AB02] these notions were generalized to the notion of infinite normal form and two more complicated cyclic extensions and Lévy-Longo trees for them were studied as examples. These examples proved to be quite complicated, because the cyclic extensions each included forms of the substitution rules

$$\begin{array}{lcl} \texttt{letrec}\,x=M,D\,\texttt{in}\,C[x] & \xrightarrow[\text{es}]{} & \texttt{letrec}\,x=M,D\,\texttt{in}\,C[M] \ ; \\ \texttt{letrec}\,x=M,y=C[x],D\,\texttt{in}\,N & \xrightarrow[\text{is}]{} & \texttt{letrec}\,x=M,y=C[M],D\,\texttt{in}\,N \ ; \\ \texttt{letrec}\,x=C[x],D\,\texttt{in}\,N & \xrightarrow[\text{cs}]{} & \texttt{letrec}\,x=C[C[x]],D\,\texttt{in}\,N \ . \end{array}$$

The problem is that with these three rules the rewrite system is no longer confluent. The classical example is:

$$\begin{array}{ccc}
\texttt{letrec}\, x = F(x)\, \texttt{in}\, x & \xrightarrow{\text{es}} & \texttt{letrec}\, x = F(x)\, \texttt{in}\, F(x) \\
\downarrow{\scriptstyle\text{cs}} & & \downarrow{\scriptstyle\text{cs}} \\
\texttt{letrec}\, x = F(F(x))\, \texttt{in}\, x & & \texttt{letrec}\, x = F(F(x))\, \texttt{in}\, F(x)
\end{array}$$

The two terms at the bottom do not have a common reduct. (Every reduct of the left term has an even number of F symbols, every reduct of the right term an odd number.)

The non-confluence causes a number of technical problems. Unfortunately, the way in which infinite normal forms were defined in [AB97, AB02] means that for every different infinite normal form that is defined, one has to deal with these problems all over. In [Blo01] an improved framework was given in which the substitution related problems are separated from the rewrite system related problems. This separation is achieved by relating the infinite normal form of a term graph to the infinite normal form of the unwinding of the term graph. The price for the separation is that we have to assume that the infinite normal form of a graph is the infinite normal form of the unwinding of that graph. In the context of modeling programming languages, this means that the letrec cannot be used to model sharing sensitive side-effects such as assignment. However, one can first model the sharing sensitive side-effects in a non-recursive setting and then use the letrec to add recursion.

Overview. After the preliminaries, we will define the notion of infinite normal form. In the next section, we extend rewrite relations and infinite normal forms from terms to graphs. Using these extensions, we then identify a class of term graph rewrite systems for which we define extensions of infinite normal forms from terms to term graphs.

2. Preliminaries

The set of terms $\mathcal{T}$ is given by:

$$\mathcal{T} ::= x \mid \lambda x.\mathcal{T} \mid F(\mathcal{T}, \cdots, \mathcal{T}) \;\; ,$$

where x ranges over a set of variables and F ranges over a set of function symbols including the constant Ω and the binary symbol @. We write $s\,t$ for $@(s,t)$. Combinatory Reduction Systems (see [KvOvR93]) are defined on terms with a single binding operator, so one may think of a CRS whenever we mention a rewrite system on terms.

We define the partial order $\leq_\Omega$ on terms as the least transitive, reflexive, compatible relation such that $\forall s \in \mathcal{T} : \Omega \leq_\Omega s$. That is, a term s is smaller than a term t if s can be obtained from t by replacing sub-terms with Ω.

We define the downward closure of a set S as:

$$\downarrow S = \{s \in \mathcal{T} \mid \exists s' \in S : s \leq_\Omega s'\} \;\; .$$

The set of possibly infinite terms $\mathcal{T}^\infty$ is the ideal completion of $(\mathcal{T}, \leq_\Omega)$. That is, a term $S \in \mathcal{T}^\infty$ is a subset of $\mathcal{T}$, such that

$$S = \downarrow S \text{ and } \forall s,t \in S : \exists u \in S : s \leq_\Omega u \wedge t \leq_\Omega u \;\; .$$

The set of terms is embedded in the set of infinite terms by means of the map

$$s \mapsto \downarrow\{s\} \;\; .$$

We define the set of cyclic terms $\mathcal{T}^\circ$ as:

$$\begin{array}{lcl}
\mathcal{T}^\circ & ::= & x \mid \lambda x.\mathcal{T}^\circ \mid F(\mathcal{T}^\circ, \cdots, \mathcal{T}^\circ) \mid \texttt{letrec}\, D\, \texttt{in}\, \mathcal{T}^\circ \;\; ; \\
D & ::= & x_1 = \mathcal{T}^\circ, \cdots, x_n = \mathcal{T}^\circ \;\; .
\end{array}$$

Given a rewrite system $(\mathcal{T}, \to)$, we denote the reflexive closure of $\to$ by $\to^{\equiv}$, the transitive reflexive closure by $\twoheadrightarrow$ and the transitive, reflexive and symmetric closure by $\twoheadleftarrow\!\!\twoheadrightarrow$. We say that $\to$ is monotonic if

$$\forall s, s', t : s \to t \wedge s \leq_\Omega s' \implies \exists t' : s' \to t' \wedge t \leq_\Omega t' \ .$$

Not every CRS is monotonic. For example, if we have the η-rule for the lambda calculus, we have $\lambda x.(\Omega\, y)\, x \underset{\eta}{\to} \Omega\, y$, but $\lambda x.(x\, y)\, x$ contains no η-redex. The problem is that the η-rule requires that a variable does not occur in a sub-term. By expanding an Ω we introduced an occurrence of a variable that destroyed the redex. To get monotonicity we therefore need rewrite rules in which no non-occurrence requirements are used. Such rewrite rules are called fully-extended in [HP96].

3. Infinite Normal Forms

In this section, we give a simple presentation of the notion of infinite normal form. This notion is an abstract version of the Böhm tree definition of Lévy (See [Lév78]). A first version of the notion of infinite normal form is presented in [AB97, AB02]. The theory was extended in [Blo01]. In these papers the theory is presented on the level of abstract reduction systems. To keep matters simple, we restrict our presentation to a subset of the theory in [Blo01] and we present the theory on the level of lambda terms.

The idea behind the notion of infinite normal form is that the result of an infinite computation is an infinite term, which is built piece-by-piece during the computation. For example, the Böhm tree of $(\lambda x.\lambda y.y\,(x\,x))\,(\lambda x.\lambda y.y\,(x\,x))$ is the limit of the bold parts of the terms in the following reduction of the term:

$$\begin{array}{l}
(\lambda x.\lambda y.y\,(x\,x))\,(\lambda x.\lambda y.y\,(x\,x)) \\
\quad \underset{\beta}{\to} \boldsymbol{\lambda y.y\ (}\ (\lambda x.\lambda y.y\,(x\,x))\,(\lambda x.\lambda y.y\,(x\,x))\ \boldsymbol{)} \\
\quad \underset{\beta}{\to} \boldsymbol{\lambda y.y\ \lambda y.y\ (}\ (\lambda x.\lambda y.y\,(x\,x))\,(\lambda x.\lambda y.y\,(x\,x))\ \boldsymbol{)} \\
\quad \vdots \\
\quad \boldsymbol{\lambda y.y\ \lambda y.y\ \lambda y.y\ (\cdots)}
\end{array}$$

If more than one computation is possible then the results over all possible reductions are gathered. The key to formalizing the notion is the *approximation function*. This function computes the *information content* of a term, which is the prefix of the eventual infinite term that has already been computed. The infinite normal form of a term is then defined as the set of all prefixes that can be computed by reducing the term. Uniqueness is an important property of the infinite normal form: it states that any two convertible terms have the same infinite normal form. Uniqueness implies that if a term is reduced then any prefix, which may be computed form a term, can also be computed from the reduct. Formally:

Definition 3.1 Given $\mathcal{T}^x \in \{\mathcal{T}, \mathcal{T}^\infty, \mathcal{T}^\circ\}$ and a rewrite system $(\mathcal{T}^x, \to)$, we say that a function $\omega : \mathcal{T}^x \to \mathcal{T}^\infty$ is an *approximation function* if

$$\forall s, t \in \mathcal{T}^x : s \to t \implies \omega(s) \subseteq \omega(t) \ .$$

Given an approximation function $\omega : \mathcal{T}^x \to \mathcal{T}^\infty$, we define the *infinite normal form* of a term as:

$$\mathrm{Inf}_\omega^{\to}(s) = \cup\{\omega(t) \mid s \twoheadrightarrow t\} \ .$$

We say that infinite normal forms are *unique* if

$$\forall s : \mathrm{Inf}_\omega(s) \in \mathcal{T}^\infty \text{ and } \forall s, t : s \twoheadleftarrow\!\!\twoheadrightarrow t : \mathrm{Inf}_\omega(s) = \mathrm{Inf}_\omega(t) \ .$$

The superscript $\rightarrow$ is sometimes omitted if it is clear which relation is meant. Due to the natural orders on terms ($\leq_\Omega$) and sets ($\subseteq$), we can talk about monotonicity of approximation function and infinite normal forms.

Example 3.2 The prefix, which is important for the Böhm Tree is the head normal form. Thus, it seems natural to define the Böhm Tree information content of a term as:

$$\omega_{\mathrm{BT}}(s) = \begin{cases} \lambda x_1 \cdots x_n.x\, \omega_{\mathrm{BT}}(s_1) \cdots \omega_{\mathrm{BT}}(s_m), & \text{if } s \equiv \lambda x_1 \cdots x_n.x\, s_1 \cdots s_m \\ \Omega & \text{, otherwise} \end{cases}$$

To show that this function is an approximation function, we use the fact that $\omega_{\mathrm{BT}}(s)$ is the normal form of s with respect to the following rewrite rules:

$$\begin{array}{ll} (\lambda x.M)\, N & \xrightarrow[\omega_{\mathrm{BT}}]{} \Omega \;\; ; \\ \lambda x.\Omega & \xrightarrow[\omega_{\mathrm{BT}}]{} \Omega \;\; ; \\ \Omega\, M & \xrightarrow[\omega_{\mathrm{BT}}]{} \Omega \;\; . \end{array}$$

We also define $\xrightarrow[\Omega]{}$ by

$$\forall M \in \mathcal{T} : \; \Omega \xrightarrow[\Omega]{} M \;\; .$$

Note that $M \leq_\Omega N$ if and only if $M \xrightarrow[\Omega]{} \!\!\!\!\rightarrow N$.

By a simple case distinction, we can show that the following diagram holds:

$$\begin{tikzcd} \cdot \arrow[r, "\Omega"] \arrow[d, "\omega_{\mathrm{BT}}"'] & \cdot \arrow[d, dashed, "\omega_{\mathrm{BT}}"] \\ \cdot \arrow[r, dashed, "\equiv", "\Omega"'] & \equiv \end{tikzcd}$$

From the diagram and the fact that $\xrightarrow[\omega_{\mathrm{BT}}]{} \subset \xrightarrow[\Omega]{}$, we conclude that ω_{BT} is monotonic with respect to $\leq_\Omega$. If $C[(\lambda x.M)\, N] \xrightarrow[\beta]{} C[M[x := N]]$ then $C[(\lambda x.M)\, N] \xrightarrow[\omega_{\mathrm{BT}}]{} C[\Omega]$. Hence , we have

$$\Omega_{\mathrm{BT}}(C[(\lambda x.M)\, N]) = \Omega_{\mathrm{BT}}(C[\Omega]) \leq_\Omega \Omega_{\mathrm{BT}}(C[M[x := N]]) \;\; .$$

The conclusion is that Ω_{BT} is an approximation function.

The following theorem formally states three important facts: confluence implies uniqueness of infinite normal forms, monotonicity of the approximation function and the rewrite relation implies monotonicity of infinite normal forms and if infinite normal forms are unique then they are infinite terms:

Theorem 3.3 Given $\mathcal{T}^x \in \{\mathcal{T}, \mathcal{T}^\infty, \mathcal{T}^\circ\}$, a rewrite system $\mathcal{R} \equiv (\mathcal{T}^x, \rightarrow)$ and an approximation function ω.

1. If $\mathcal{R}$ is confluent then Inf_ω infinite normal forms are unique.

2. If $\rightarrow$ and ω are monotonic then Inf_ω is monotonic.

3. If Inf_ω infinite normal forms are unique then $\forall s \in \mathcal{T}^x : \; \mathrm{Inf}_\omega(s) \in \mathcal{T}^\infty$.

1. Given $s \twoheadleftarrow\!\!\twoheadrightarrow t$ and $a \in \mathrm{Inf}_\omega(s)$. We can find s', such that $s \twoheadrightarrow s'$ and $a \in \omega(s')$. By confluence we can find t', such that $s' \twoheadrightarrow s'$ and $t \twoheadrightarrow t'$. Because ω is an approximation function, we have $\omega(s') \subseteq \omega(t')$. Thus, we have:

$$a \in \omega(s') \subseteq \omega(t') \subseteq \mathrm{Inf}_\omega(t) \;\; .$$

2. Given $s \leq \Omega t$ and $a \in \mathrm{Inf}_\omega(s)$. We can find s', such that $s \twoheadrightarrow s'$ and $a \in \omega(s')$. Due to monotonicity of $\rightarrow$ we can find t', such that $t \twoheadrightarrow t'$ and $s' \leq_\Omega t'$. Due to monotonicity of ω, we have $\omega(s') \subseteq \omega(t')$. Thus, we have:

$$a \in \omega(s') \subseteq \omega(t') \subseteq \mathrm{Inf}_\omega(t) \ .$$

3. For any term $\forall s \in \mathcal{T}^x$, we must show that the set $\mathrm{Inf}_\omega(s)$ is an ideal:
   - Given $a \in \mathrm{Inf}_\omega(s)$ and $a' \leq \Omega a$. We can find t, such that $s \twoheadrightarrow t$ and $a \in \omega(t)$. Because $\omega(t)$ is downward closed by definition, we have $a' \in \omega(t)$ and hence $a' \in \mathrm{Inf}_\omega(s)$.
   - Given $a_1, a_2 \in \mathrm{Inf}_\omega(s)$, we can find t, such that $s \twoheadrightarrow t$ and $a_1 \in \omega(t)$. Because of uniqueness, $\mathrm{Inf}_\omega(s) = \mathrm{Inf}_\omega(t)$, so we can find t', such that $t \twoheadrightarrow t'$ and $a_2 \in \omega(t')$. Because ω is an approximation function, we also have that $a_1 \in \omega(t')$. Because $\omega(t')$, we can then find $a_3 \in \omega(t')$, such that $a_1 \leq_\Omega a_3 \wedge a_2 \leq_\Omega a_3$. We also have that $a_3 \in \mathrm{Inf}_\omega(s)$.

□

Both uniqueness and monotonicity are used in the extensions of infinite normal forms from terms to graphs. If all of these properties hold then we talk about a regular rewriting system with approximations:

Definition 3.4 A *Regular Rewriting System with Approximations* (RRSA) is a structure $(\mathcal{T}, \rightarrow, \omega)$, where $(\mathcal{T}, \rightarrow)$ is a rewrite system, $\rightarrow$ is monotonic and ω is a monotonic approximation function, such that Inf_ω is unique and monotonic.

Not all of these properties are needed for every stage in the development of infinite normal forms for infinite terms and term graphs, but having all of them simplifies the presentation.

Below, we define the unwinding of a term graph as an infinite normal form of the rewrite system $(\mathcal{T}^\circ, \underset{\mathrm{es}}{\longrightarrow})$. Because the unwinding is important in the remainder, we use a different notation for the infinite normal form:

Definition 3.5 Given the rewrite system $(\mathcal{T}^\circ, \underset{\mathrm{es}}{\longrightarrow})$. Let $\mathrm{nf}_{\omega_{\mathrm{unw}}}(M)$ denote the normal form of M with respect to the rewrite rule

$$\mathtt{letrec}\, x_1 = M_1, \cdots, x_n = M_n \,\mathtt{in}\, M \underset{\omega_{\mathrm{unw}}}{\longrightarrow} M[x_1 := \Omega, \cdots, x_n := \Omega]$$

The function $\omega_{\mathrm{unw}} : \mathcal{T}^\circ \rightarrow \mathcal{T}^\infty$ is defined by $\omega_{\mathrm{unw}}(M) = \downarrow \{\mathrm{nf}_{\omega_{\mathrm{unw}}}(M)\}$. The unwinding of a term M is given by:

$$\mathrm{Unw}(M) = \mathrm{Inf}^{\overrightarrow{\mathrm{es}}}_{\omega_{\mathrm{unw}}}(M) \ .$$

We use $M =_{\mathrm{unw}} N$ as shorthand for $\mathrm{Unw}(M) = \mathrm{Unw}(N)$. It is easy to verify that $\mathrm{nf}_{\omega_{\mathrm{unw}}}$ is an approximation function. Because $\underset{\mathrm{es}}{\longrightarrow}$ is confluent it is also easy to prove that Unw is a unique infinite normal form by applying Thm. 3.3.

4. Infinitary Extension

One of the motivations to study infinitary rewriting is to give the semantics of graph rewriting by means of infinitary rewriting. For this purpose Corradini defined a rewriting system on infinite terms that performs a complete development of an infinite set of redexes in one step [Cor93]. We use a liberal extension of his definition to express the requirements under which we can extend an infinite normal form definition from term rewriting to graph rewriting. Our extension lifts any rewrite relation from terms to infinite terms, by stating that S rewrites to T if every term in S can be extended to another term in S, which in turn rewrites to a term in T. Moreover, every term in T is the prefix of a term in T, which can be obtained by rewriting a term in S. Corradini's definition extends only a specific relation and requires that every prefix of S rewrites to a prefix of T. Formally:

Definition 4.1 The rewrite extension operator $[\cdot\rangle : 2^{\mathcal{T}\times\mathcal{T}} \to 2^{\mathcal{T}^\infty\times\mathcal{T}^\infty}$ is given by

$$\forall S,T \in \mathcal{T}^\infty, \to\subseteq \mathcal{T}\times\mathcal{T} :\ S[\to\rangle T \iff \left\{ \begin{array}{ll} & \forall s \in S : \exists s' \in S, t' \in T : s \leq_\Omega s' \wedge s' \to t' \\ \wedge & \forall t \in T : \exists s' \in S, t' \in T : s' \to t' \wedge t \leq_\Omega t' \end{array} \right.$$

The transitive reflexive closure of $[\cdot\rangle$ is $[\cdot\rangle\rangle$. Note that if a 'spanning' subset of S rewrites to a 'spanning' subset of T, we have that S rewrites to T:

Proposition 4.2 Given $S,T \in \mathcal{T}^\infty$, $\to\subseteq 2^{\mathcal{T}\times\mathcal{T}}$, an index set I and $s_i, t_i \in \mathcal{T}$ such that $s_i \to t_i$ for $i \in I$.

$$S = \downarrow \{s_i \mid i \in I\} \text{ and } T = \downarrow \{t_i \mid i \in I\} \implies S[\to\rangle T\ .$$

This simple observation is very useful for showing that an infinite term rewrites to another infinite term. The following example illustrates a few possibilities of the $[\cdot\rangle$ operator:

Example 4.3 Let us consider the TRS

$$A(X) \to X \quad B(X) \to X\ .$$

We use the following notation:

$$\begin{array}{lll} A^0(x) = x; & B^0(x) = x; & AB^0(x) = x; \\ A^{n+1}(x) = A(A^n(x)); & B^{n+1}(x) = B(B^n(x)); & AB^{n+1}(x) = A(B(AB^n(x))); \\ A^\omega = \{A^n \mid n \in \mathbb{N}\}; & B^\omega = \{B^n \mid n \in \mathbb{N}\}; & AB^\omega = \downarrow\{AB^n \mid n \in \mathbb{N}\}. \end{array}$$

We then have:

$$A(B(AB^\omega))[\to\rangle A(AB^\omega), \text{ because } A(B(AB^n(\Omega)) \to A(AB^n(\Omega))\ ;$$

$$AB^\omega[\twoheadrightarrow\rangle A^\omega, \text{ because } AB^n(\Omega)\twoheadrightarrow A^n(\Omega)\ ;$$

$$A^\omega[\twoheadrightarrow\rangle\Omega, \text{ because } A^n(\Omega)\twoheadrightarrow\Omega\ ;$$

$$AB^\omega[\twoheadrightarrow\rangle B^\omega[\twoheadrightarrow\rangle\Omega\ .$$

One may ask the question if confluence of $\to$ implies confluence of $[\twoheadrightarrow\rangle$. The answer is no, one will have to impose some restrictions. For example, with the confluent rule

$$A(A(X)) \underset{2}{\to} X$$

we get both

$$A^\omega[\underset{2}{\twoheadrightarrow}\rangle\Omega \text{ and } A^\omega[\underset{2}{\twoheadrightarrow}\rangle A(\Omega)\ .$$

The $[\cdot\rangle$ operator extends a rewrite relation form terms to infinite term. If on the terms, we have an infinite normal form Inf_ω defined then we want to extend the infinite normal form to infinite terms as well. The simplest way to do this is to extend the function Inf_ω. Because infinite terms are sets of terms and the co-domain of Inf_ω is sets of terms already, we only extend the domain:

Definition 4.4 Given a RRSA $(\mathcal{T}, \to, \omega)$, the *direct extension* of Inf_ω is defined as:

$$\mathrm{Inf}_\omega^\infty(S) = \cup\{\mathrm{Inf}_\omega(s) \mid s \in S\}\ .$$

The direct extension is a proper extension in the sense that for finite terms Inf_ω and $\mathrm{Inf}_\omega^\infty$ yield the same result:

Proposition 4.5 Given a RRSA $(\mathcal{T}, \rightarrow, \omega)$, we have that:

$$\forall t \in \mathcal{T} : \mathrm{Inf}_\omega(t) = \mathrm{Inf}_\omega^\infty(\downarrow \{t\}) \ .$$

We distinguish two cases:

"$\subseteq$": Follows from the fact that $t \in \downarrow \{t\}$.

"$\supseteq$": From monotonicity if follows that for every $s \in \downarrow \{t\}$, we have that

$$\mathrm{Inf}_\omega(s) \subseteq \mathrm{Inf}_\omega(t) \ .$$

Hence, we have that

$$\mathrm{Inf}_\omega^\infty(\downarrow \{t\}) \subseteq \mathrm{Inf}_\omega(t) \ .$$

□

The direct extension is also unique in the following sense:

Proposition 4.6 Given a RRSA $(\mathcal{T}, \rightarrow, \omega)$, we have that:

$$\forall S, T \in \mathcal{T}^\infty : \ S[\twoheadrightarrow\rangle T \implies \mathrm{Inf}_\omega^\infty(S) = \mathrm{Inf}_\omega^\infty(T) \ .$$

We distinguish two cases:

"$\subseteq$": Given $s \in S$, we can find $s' \in S, t' \in T$, such that $s \leq s'$ and $s' \twoheadrightarrow t'$. By monotonicity and uniqueness respectively we then have that

$$\mathrm{Inf}_\omega(s) \subseteq \mathrm{Inf}_\omega(s') \text{ and } \mathrm{Inf}_\omega(s') = \mathrm{Inf}_\omega(t') \ .$$

From these two statements and the definition of $\mathrm{Inf}_\omega^\infty$ we can derive that

$$\mathrm{Inf}_\omega^\infty(S) \subseteq \mathrm{Inf}_\omega^\infty(T) \ .$$

"$\supseteq$": Similar to the previous case.

□

Although the direct extension has the right properties (see the previous two propositions), it isn't satisfactory in the sense that it is not defined by means of an approximation function. Nevertheless, we need it to prove that the approximation function for infinite terms, we will define next, yields unique infinite normal forms.

Definition 4.7 Given a RRSA $(\mathcal{T}, \rightarrow, \omega)$, we define the *derived extension* of Inf_ω with respect to $\Rightarrow \subseteq \mathcal{T} \times \mathcal{T}$ as:

$$\omega^\infty(S) = \downarrow \{\omega(s) \mid s \in S\}$$

$$\mathrm{Inf}_{\omega^\infty}^{\Rightarrow}(S) = \cup\{\omega^\infty(T) \mid S[\Rightarrow\rangle\rangle T\}$$

We want to prove that the direct extension is equal to the derived extension. To do so, we need to lift a reduction on terms to a reduction on infinite terms. This is possible for monotone reduction relations:

Lemma 4.8 Given a rewrite system $(\mathcal{T}, \rightarrow)$. If $\rightarrow$ is monotone then

$$\forall s, t \in \mathcal{T}, S \in \mathcal{T}^\infty : s \rightarrow t \wedge s \in S \implies \exists T \in \mathcal{T}^\infty : S[\rightarrow\rangle T \wedge t \in T \ .$$

Given $s, t \in \mathcal{T}, S \in \mathcal{T}^\infty$, such that $s \to t$ and $s \in S$. Because S is a countable set and an ideal, we can find a sequence

$$s \equiv s_0 \leq_\Omega s_1 \leq_\Omega \cdots, \text{ such that } S = \downarrow \{s_i \mid i \geq 0\} \ .$$

Because $\to$ is monotonic we can find a sequence t_i, such that $s_i \to t_i$ and $t_i \leq_\Omega t_{i+1}$. We then have that

$$S \; [\to\rangle \downarrow \{t_i \mid i \geq 0\} \ .$$

□

With this lemma, we can prove the equality of the two extensions:

Theorem 4.9 Given a RRSA $(\mathcal{T}, \to, \omega)$, we have that

$$\forall S \in \mathcal{T}^\infty \mathrm{Inf}^\infty_\omega(S) = \mathrm{Inf}^{\twoheadrightarrow}_{\omega^\infty}(S) \ .$$

We distinguish two cases:

"$\supseteq$": Given $a \in \mathrm{Inf}^{\twoheadrightarrow}_{\omega^\infty}(S)$. According to Def. 4.7 we can find a sequence

$$S \equiv S_0 [\twoheadrightarrow\rangle S_1 [\twoheadrightarrow\rangle \cdots S_n, \text{ such that } \exists s_n \in S_n : a \in \omega(s_n) \ .$$

From $a \in \omega(s_n)$ we get that $a \in \mathrm{Inf}_\omega(s_n)$. By applying Def. 4.1 we can find $s_{n-1} \in S_{n-1}, s'_n \in S_n$, such that $s_n \leq_\Omega s'_n$ and $s'_n \twoheadleftarrow s_{n-1}$. From monotonicity and uniqueness it follows that $\mathrm{Inf}_\omega(s_n) \subseteq \mathrm{Inf}_\omega(s'_n)$ and $\mathrm{Inf}_\omega(s'_n) = \mathrm{Inf}_\omega(s_{n-1})$, so $\mathrm{Inf}_\omega(s_n) \subseteq \mathrm{Inf}_\omega(s_{n-1})$. By repeating this argument we can find $s_0 \in S_0$ such that $\mathrm{Inf}_\omega(s_n) \subseteq \mathrm{Inf}_\omega(s_0)$. We conclude that

$$a \in \mathrm{Inf}_\omega(s_n) \subseteq \mathrm{Inf}_\omega(s_0) \subseteq \mathrm{Inf}^\infty_\omega(S_0) \ .$$

"$\subseteq$": Given $a \in \mathrm{Inf}^\infty_\omega(S)$. We can find an $s \in S$ such that $a \in \mathrm{Inf}_\omega(s)$. Hence, we can find $t \in \mathcal{T}$, such that $s \twoheadrightarrow t$ and $a \in \omega(t)$. By repeatedly applying Lemma 4.8, we can find $T \in \mathcal{T}^\infty$, such that $t \in T$ and $S \; [\to\rangle\!\rangle \; T$. Because $[\to\rangle \subseteq [\twoheadrightarrow\rangle$ and $a \in \omega(t) \subseteq \omega^\infty(T)$, we have

$$a \in \mathrm{Inf}^{\twoheadrightarrow}_{\omega^\infty}(S) \ .$$

□

A very easy corollary of this theorem is that the derived infinite normal form is unique.

Corollary 4.10 Given a RRSA $(\mathcal{T}, \to, \omega)$, we have that $\mathrm{Inf}^{\twoheadrightarrow}_{\omega^\infty}$ infinite normal forms are unique.

5. Cyclic Extension

A cyclic extension of a rewrite system on terms is nothing but a rewrite system on cyclic terms that has somehow been derived from the rewrite system on terms. For the purpose of this paper it is not important how this derivation was done. Only the properties of the derivation matter. The most important property is infinitary soundness, which says that any reduction on cyclic terms can be projected to a reduction on the unwindings of the terms:

Definition 5.1 A cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$ of a rewrite system $(\mathcal{T}, \xrightarrow[R]{})$ is infinitarily sound if

$$\forall M, N : M \xrightarrow[R^\circ]{} N \implies \mathrm{Unw}(M) [\underset{R}{\twoheadrightarrow}\rangle \mathrm{Unw}(N) \ .$$

One way of deriving a cyclic extension is to begin with modifying the right-hand sides of the given rewrite rules to use letrecs for substitutions. Next, rewrite rules that do not modify the unwinding may be added. Finally, some completion must be done. The simple cyclic extension of the lambda calculus in the introduction (1.1) was derived in this way. If a cyclic extension of an orthogonal CRS is derived in this way then the following theorem, which was proven in [Blo01], might be useful to show infinitary soundness:

Theorem 5.2 Given a cyclic extension $\mathcal{R}^\circ \equiv (\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$ of an orthogonal CRS $(\mathcal{T}, \xrightarrow[R]{})$. The cyclic extension is infinitarily sound if

$$\xrightarrow[R^\circ]{} \subseteq =_{\mathrm{unw}} \cup (=_{\mathrm{unw}} \circ \xrightarrow[R]{} \circ =_{\mathrm{unw}}) \ .$$

Through the unwinding of a cyclic term, we can give a direct extension of an infinite normal form from terms to cyclic terms:

Definition 5.3 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$ and a cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$, we define the *direct extension* of Inf_ω as

$$\mathrm{Inf}^\circ_\omega(M) = \mathrm{Inf}^\infty_\omega(\mathrm{Unw}(M)) \ .$$

This extension is unique in the following sense:

Proposition 5.4 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$ and an infinitarily sound cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$, we have

$$\forall M, N \in \mathcal{T}^\circ : \ M \xrightarrow[R^\circ]{} N \implies \mathrm{Inf}^\circ(M) = \mathrm{Inf}^\circ(N) \ .$$

Follows from infinitary soundness and Prop. 4.6. □

A derived extension is also possible:

Definition 5.5 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$ and an infinitarily sound cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$, the *derived extension* of Inf_ω is defined by the approximation function

$$\omega^\circ(M) = \downarrow \{\omega(s) \mid s \in \mathrm{Unw}(M)\} \ .$$

To prove that the derived extension is equal to the direct extension and hence unique, we will need a notion of completeness. The perfect situation would be that if an approximation s of the unwinding of a cyclic term M rewrites to a term t then M rewrites to a cyclic term N such that t is an approximation of the unwinding of N. However, we cannot expect this due to the fact that a rewrite step on a cyclic term often corresponds to more than one step on an approximation. The matter of completeness is discussed in depth in [Blo01]. In this paper, we merely present the solution.

Definition 5.6 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$. A cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$ is *complete up to information content* if $\forall s, t \in \mathcal{T}, M \in \mathcal{T}^\circ$:

$$s \in \mathrm{Unw}(M) \wedge s \xrightarrow[R]{}\!\!\!\!\rightarrow t \implies \exists N \in \mathcal{T}^\circ, t' \in \mathrm{Unw}(N) : M \xrightarrow[R^\circ]{}\!\!\!\!\rightarrow N \wedge \omega(t) \subseteq \omega(t').$$

We can make the notion of completeness stronger by replacing $\omega(t) \subseteq \omega(t')$ with $t \xrightarrow[R]{}\!\!\!\!\rightarrow t'$. The resulting stronger notion is usually easier to prove, but it doesn't always hold and the weak notion is sufficient to imply that the derived and direct infinite normal forms are the same.

Definition 5.7 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$. A cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$ is regular if it is infinitarily sound and complete up to information content.

Theorem 5.8 Given a RRSA $\mathcal{R} \equiv (\mathcal{T}, \xrightarrow[R]{}, \omega)$ and a regular cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$, we have that

$$\forall M \in \mathcal{T}^\circ : \mathrm{Inf}_{\omega^\circ}(M) = \mathrm{Inf}^\circ_\omega(M) \ .$$

We distinguish two cases:

"$\subseteq$": From infinitary soundness it follows that $\mathrm{Inf}_{\omega^\circ}(M) \subseteq \mathrm{Inf}_{\omega^\infty}(\mathrm{Unw}(M))$. Hence, by Thm. 4.9 it follows that

$$\mathrm{Inf}_{\omega^\circ}(M) \subseteq \mathrm{Inf}^\circ_\omega(M) \ .$$

"$\supseteq$": Given $a \in \mathrm{Inf}^\circ_\omega(M)$. By definition $\mathrm{Inf}^\circ_\omega(M) = \mathrm{Inf}^\infty_\omega(\mathrm{Unw}(M)) = \cup\{\mathrm{Inf}(s) \mid s \in \mathrm{Unw}(M)\}$, so we can find $s \in \mathrm{Unw}(M)$ such that $a \in \mathrm{Inf}_\omega(s)$. Thus, we can also find $t \in \mathcal{T}$ such that $s \twoheadrightarrow_R t$ and $a \in \omega(t)$. Due to completeness, we can find an $N \in \mathcal{T}^\circ, t' \in \mathrm{Unw}(N)$ such that $M \twoheadrightarrow_{R^\circ} N$ and $\omega(t) \subseteq \omega(t')$. Hence:

$$a \in \omega(t) \subseteq \omega(t') \subseteq \omega^\circ(N) \subseteq \mathrm{Inf}_{\omega^\circ}(M) \ .$$

□

Uniqueness of the derived infinite normal form is an easy corollary of this theorem.

Corollary 5.9 Given a RRSA $\mathcal{R} \equiv (\mathcal{T}, \xrightarrow[R]{}, \omega)$ and a regular cyclic extension $(\mathcal{T}^\circ, \xrightarrow[R^\circ]{})$, we have that $\mathrm{Inf}_{\omega^\circ}$ infinite normal forms are unique.

5.1 Syntactic continuity and congruence relations.
This section gives a brief sketch of how congruence results can be lifted from term rewriting to cyclic term rewriting. Due to space limitations the full details cannot be included here. They can be found in [Blo01].

To prove that $\mathrm{Inf}(M) = \mathrm{Inf}(N) \implies \mathrm{Inf}(C[M]) = \mathrm{Inf}(C[N])$ Lévy used the notion of syntactic continuity. Syntactic continuity expresses that to compute the infinite normal form of a context filled with a term , everything you need to know about that term is contained in the infinite normal form of that term:

Definition 5.10 Given a rewrite system $(\mathcal{T}, \xrightarrow[R]{})$ and an approximation function $\omega : \mathcal{T} \to \mathcal{T}^\infty$, we have *syntactic continuity* of Inf_ω if

$$\mathrm{Inf}_\omega(C[s]) = \cup\{\mathrm{Inf}_\omega(C[a]) \mid a \in \mathrm{Inf}_\omega(s)\} \ .$$

To extend congruence results from terms to cyclic terms, we will also need the notion of *substitutive continuity*, which expresses that to compute the infinite normal form of a substitution applied to a term, everything you need to know about that term is contained in the infinite normal form of that term:

Definition 5.11 Given a rewrite system $(\mathcal{T}, \xrightarrow[R]{})$ and an approximation function $\omega : \mathcal{T} \to \mathcal{T}^\infty$, we have *substitutive continuity* of Inf_ω if

$$\mathrm{Inf}_\omega(s\sigma) = \cup\{\mathrm{Inf}_\omega(a\sigma) \mid a \in \mathrm{Inf}_\omega(s)\} \ .$$

Note that, if the β-rule is present substitutive continuity follows immediately from syntactic continuity due to the fact that

$$\mathrm{Inf}_\omega(s[x_1 := t_1, \cdots, x_n := t_n]) = \mathrm{Inf}_\omega(C[s]) \ ,$$
$$\text{where } C \equiv (\lambda x_1 \cdots x_n.\Box)\, t_1 \cdots t_n \ .$$

Given substitutive and syntactic continuity of a suitable CRS, we can prove that any regular cyclic extension has syntactic continuity:

Theorem 5.12 Given a RRSA $(\mathcal{T}, \xrightarrow[R]{}, \omega)$, where $(\mathcal{T}, \xrightarrow[R]{})$ is a fully-extended orthogonal CRS, and a regular cyclic extension $(\mathcal{T}^{\circ}, \xrightarrow[R^{\circ}]{})$, we have that syntactic and substitutive continuity of Inf_{ω} imply syntactic continuity of $\mathrm{Inf}^{\circ}_{\omega}$.

6. Conclusion

We have presented a framework for lifting infinite normal form definitions from term rewrite systems to term graph rewrite systems. The non-confluence problems, which had to be dealt with for proving uniqueness of infinite normal forms in [AI97] are now dealt with in a proof of infinitary soundness. The advantage is that in a proof of infinitary soundness we can use known results about rewrite rules that preserve the unwinding of a term. The framework does not supersede the notion of *skew confluence* ([AI97]). This notion is still useful to prove results about rewrite rules that preserve the unwinding of a term.

Acknowledgments We thank Vincent van Oostrom for his help on Combinatory Reduction Systems and Jaco van de Pol for proof reading this paper.

References

[AB97] Z. M. Ariola and S. Blom. Cyclic lambda calculi. In Abadi and Ito [AI97], pages 77–106.

[AB02] Zena M. Ariola and Stefan Blom. Skew confluence and the lambda calculus with letrec. *Annals of Pure and Applied Logic*, 117(1-3):95–168, october 2002.

[AI97] Martín Abadi and Takayasu Ito, editors. *Theoretical Aspects of Computer Software*, volume 1281 of *Lecture Notes in Computer Science*. Springer Verlag, September 1997.

[AK97] Z. M. Ariola and J. W. Klop. Lambda calculus with explicit recursion. *Information and computation*, 139(2):154–233, December 1997.

[Blo01] S.C.C. Blom. *Term Graph Rewriting - syntax and semantics*. PhD thesis, Vrije Universiteit Amsterdam, 2001.

[Cor93] Andrea Corradini. Term rewriting in ct_{σ}. In *Proc. CAAP '93*, volume 668 of *Lecture Notes in Computer Science*, pages 468–484. Springer, Berlin, 1993.

[HP96] Michael Hanus and Christian Prehofer. Higher-order narrowing with definitional trees. In Harald Ganzinger, editor, *RTA*, volume 1103 of *Lecture Notes in Computer Science*, pages 138–152. Springer, 1996.

[KvOvR93] J. W. Klop, V. van Oostrom, and F. van Raamsdonk. Combinatory reduction systems: Introduction and survey. *Theoretical Computer Science*, 121(1-2):279–308, 1993. A Collection of Contributions in Honour of Corrado Böhm on the Occasion of his 70th Birthday, guest eds. M. Dezani-Ciancaglini, S. Ronchi Della Rocca and M. Venturini-Zilli.

[Lév78] J.-J. Lévy. *Réductions Correctes et Optimales dans le Lambda-Calcul*. PhD thesis, Universite Paris VII, October 1978.