Counterexamples in Infinitary Rewriting with Non-Fully-Extended Rules

Jeroen Ketema

Research Institute of Electrical Communication, Tohoku University
2-1-1 Katahira, Aoba-ku, Sendai 980-8577, Japan

Abstract

We show counterexamples exist to confluence modulo hypercollapsing subterms, fair normalisation, and the normal form property in orthogonal infinitary higher-order rewriting with non-fully-extended rules. This sets these systems apart from both fully-extended and finite systems, where no such counterexamples are possible.

Key words: programming calculi, infinitary rewriting, higher-order rewriting

1 Introduction

Infinitary higher-order rewriting [8,9] extends higher-order rewriting with infinite terms and transfinite reductions. It can be used to model lazy functional languages, which allow for potentially infinite data structures. The theory developed thus far requires both *orthogonality* and *fully-extendedness*. Hence, it is neither allowed to have overlap between rules, nor to have rules which check for the non-occurrence of bound variables; an example of a non-fully-extended rule is η-rule from λ-calculus:

$$\lambda x.Mx \rightarrow_\eta M\,.$$

This rule may only be applied in case the variable x does not occur in M, i.e. we must check for the non-occurrence of x in M.

It is an open problem [9,11,10,6] whether the results obtained for fully-extended, orthogonal higher-order systems carry over to non-fully-extended systems.

Email address: jketema@nue.riec.tohoku.ac.jp (Jeroen Ketema).

As it turns out, this is in general not the case: Below we provide counterexamples to confluence modulo hypercollapsing subterms, fair normalisation [1], and the normal form property.

The existence of the counterexamples is remarkable, as the results in infinitary rewriting derive from similar results in finite rewriting, where fully-extendedness irrelevant. However, *non-confluence* of infinitary orthogonal systems [4,3] comes into play here; consider the rewrite rule

$$g(Z, Z') \to Z$$

and the infinite terms $s_{x,y}$ and t_x represented by the recursive equations $s_{x,y} = g(g(s_{x,y}, y), x)$ and $t_x = g(t_x, x)$, where x and y are arbitrary variables. Contracting in an outside-in fashion all redexes in $s_{x,y}$ of the form $g(\dots, y)$, resp. of the form $g(\dots, x)$, we obtain t_x, resp. t_y, in an infinite number of steps. These latter terms only reduce to themselves and do not have a common reduct. Our counterexamples below depend on x occurring in t_x but not in t_y and x being non-removable from t_x by reduction.

The rule presented above and all other rules that occur below are cast in the format of infinitary Combinatory Reduction Systems (iCRSs), the only currently available format for infinitary higher-order rewriting and an extension of the finite format of Combinatory Reduction Systems (CRSs) [12]. However, once developed, the counterexamples are likely to carry over to other formats of infinitary higher-order rewriting that allow for non-fully-extended rules, as any differences compared to iCRSs will likely be mostly of a syntactic nature.

The paper proceeds as follows: In Section 2, we give some preliminaries. In Section 3, we provide counterexamples to two intermediate properties which in the fully-extended case are key in obtaining confluence modulo hypercollapsing subterms, fair normalisation, and the normal form property. In Section 4, we show the failure of the central properties and, in Section 5, we conclude and mention some directions for further research.

2 Preliminaries

We summarise some basic facts concerning (higher-order) infinitary rewriting [3,4,2,8,9,11,10]. Throughout, we denote the first infinite ordinal by ω, and arbitrary ordinals by α, β, γ, and so on.

[1] Counterexamples to needed(-fair) and outermost-fair normalisation [6,11] already occur in the finite case [15] and apply equally well in our setting.

Terms and Substitutions. Assume a signature Σ, each element of which has finite arity. Also assume a countably infinite set of variables and, for each finite arity, a countably infinite set of meta-variables.

Terms in infinitary rewriting are usually introduced by defining an appropriate metric over the finite terms and employing metric completion [4,8]. Here, we give the shorter, but equivalent, definition from [9]:

Definition 2.1 *The set of* meta-terms *is defined coinductively by the following rules, where s and $s_1, \ldots, s_n$ are again meta-terms: (1) each variable x is a meta-term, (2) if x is a variable, then $[x]s$ is a meta-term, (3) if Z is a meta-variable of arity n, then $Z(s_1, \ldots, s_n)$ is a meta-term, and (4) if $f \in \Sigma$ is of arity n, then $f(s_1, \ldots, s_n)$ is a meta-term.*

The set of finite *meta-terms, a subset of the set of meta-terms, is inductively defined by the above rules. A* term *is a meta-term without meta-variables and a* context *is a meta-term over $\Sigma \cup \{\Box\}$.*

We consider (meta-)terms modulo α-equivalence. A meta-term of the form $[x]s$ is called an *abstraction*. An occurrence of a variable x is *bound* in s if x occurs in a subterm of the form $[x]t$; x is *free* otherwise. Meta-terms with meta-variables occur only in rewrite rules, where meta-variables assume the rôle variables have in the first-order case; rewriting is defined over terms.

The set of *positions* [8] of a meta-term s is a set of *finite* strings over $\mathbb{N}$ with each string corresponding to the 'location' of a subterm of s. If p is a position of s, then $s|_p$ is the *subterm of s at* position p (e.g. $f(s_1, \ldots, s_n)|_i = s_i$). The length of p is denoted $|p|$. The concatenation of p and q is denoted $p \cdot q$.

A *valuation*, the iCRS counterpart of a substitution and denoted $\bar{\sigma}$, is defined by interpreting the CRS rules for valuations [12] coinductively. A valuation substitutes terms for meta-variables. In CRSs, applying a valuation to a meta-term yields a unique term; this is no longer the case in the current setting [8]. However, applying a valuation to a meta-term yields a unique term in case the meta-term satisfies the so-called finite chains property [8]:

Definition 2.2 *Let s be a meta-term. A* chain *in s is a sequence of (context, position)-pairs $(C_i[\Box], p_i)_{i<\alpha}$, with $\alpha \leq \omega$, such that for each $(C_i[\Box], p_i)$ there is a term t_i with $C_i[t_i] = s|_{p_i}$ and $p_{i+1} = p_i \cdot q$ where q is the position of the hole in $C_i[\Box]$. A* chain of meta-variables *in s is a chain such that for each $i < \alpha$ it holds that $C_i[\Box] = Z(t_1, \ldots, t_n)$ with $t_j = \Box$ for exactly one $1 \leq j \leq n$.*

The meta-term s is said to satisfy the finite chains property *if no infinite chain of meta-variables occurs in s.*

Note that $\Box$ only occurs in $C_i[\Box]$ if $i + 1 < \alpha$, otherwise $C_i[\Box] = s|_{p_i}$. Finite meta-terms always satisfy the finite chains property. The meta-term

$$[x_1]Z_1([x_2]Z_2(\ldots [x_n]Z_n(\ldots)))$$

also satisfies the finite chains property, whereas

$$Z(Z(\ldots Z(\ldots)))$$

does not.

Rewriting. Recall that a *pattern* is a finite meta-term in which each meta-variable has distinct bound variables as arguments and that a meta-term is *closed* if all variables occur bound [12]. We define rewrite rules and iCRSs:

Definition 2.3 *A* rewrite rule *is a pair of closed meta-terms (l, r), denoted $l \to r$, with l finite and r satisfying the finite chains property, such that l is a pattern of the form $f(s_1, \ldots, s_n)$ and such that all meta-variables that occur in r also occur in l.*

An infinitary Combinatory Reduction System (iCRS) *is a pair $\mathcal{C} = (\Sigma, R)$ with Σ a signature and R a set of rewrite rules.*

Orthogonality, i.e. the combination of left-linearity and non-overlap of rules, can be defined as in the case of CRSs [12], as left-hand sides of rewrite rules are finite. Besides orthogonality, we employ two other properties of rewrite rules. First, a rule is *collapsing* if its right-hand side has a meta-variable at the root. Second, a pattern is *fully-extended*, if, for each meta-variable Z and abstraction $[x]s$ with an occurrence of Z in its scope, x is an argument of that occurrence of Z; a rule is *fully-extended* if its left-hand side is [1,14].

As an example, consider the β- and η-rule from λ-calculus, which can, resp., be modelled by the following iCRS rules:

$$\begin{aligned} \texttt{app}(\texttt{lam}([x]Z(x)), Z') &\to Z(Z') \\ \texttt{lam}([x]\texttt{app}(Z, x)) &\to Z \end{aligned}$$

Both rules are collapsing, but together they do not form an orthogonal iCRS (although the rules separately do). The β-rule is fully-extended, as x occurs as an argument of Z in the left-hand side of the rule; the η-rule is *not* fully-extended, as x does not occur as an argument of Z in the left-hand side. Note that the non-occur check is implicit in the encoding of the η-rule.

We next define rewrite steps:

Definition 2.4 *A* rewrite step *is a pair terms (s, t) denoted $s \to t$ and adorned with a context $C[\Box]$, a rewrite rule $l \to r$, and a valuation $\bar{\sigma}$ such* *that* $s = C[\bar{\sigma}(l)]$ *and* $t = C[\bar{\sigma}(r)]$. *The term* $\bar{\sigma}(l)$ *is called a* redex. *The redex* occurs *at depth* $|p|$ *in* s, *where* p *is the position of the hole in* $C[\Box]$.

Both $\bar{\sigma}(l)$ and $\bar{\sigma}(r)$ are well-defined, as left-hand and right-hand sides of rewrite rules satisfy the finite chains property (left-hand sides do so as they are finite). We call a redex or a rewrite step *collapsing* if the employed rewrite rule is.

We can now define transfinite reductions and strongly convergent reductions:

Definition 2.5 *A* transfinite reduction *is a sequence of terms* $(s_\beta)_{\beta<\alpha}$, *adorned with a rewrite step* $s_\beta \to s_{\beta+1}$ *for each* $\beta + 1 < \alpha$. *Let* d_β *denote the depth of the redex contracted in* $s_\beta \to s_{\beta+1}$, *the reduction is* strongly convergent, *denoted* $s_0 \twoheadrightarrow s_\alpha$, *if* α *is a successor ordinal and if* s_β *converges to* s_γ *and* d_β *tends to infinity as* β *approaches* γ *from below for every limit ordinal* $\gamma < \alpha$.

All key properties mentioned below concern strongly convergent reductions. The intermediate properties require two other types of reductions: A transfinite reduction $(s_\beta)_{\beta<\alpha}$ of limit ordinal length with every initial sequence strictly shorter than α strongly convergent is called *perpetual*, resp. *hypercollapsing*, if an infinite number of steps, resp. an infinite number of collapsing steps, occurs at the root.

We employ three notions related to strongly convergent reductions: A term s is a *normal form*, resp. a *head normal form*, if no redex occurs in s, resp. if s does not reduce to a redex by a strongly convergent reduction. Moreover, s is *hypercollapsing* if for all $s \twoheadrightarrow t$ there exists a $t \twoheadrightarrow t'$ such that t' is a collapsing redex. Observe that a head normal form can never have a hypercollapsing reduction starting from it. Residuals across strongly convergent reductions are defined as usual [8].

3 Failure of Intermediate Properties

In the setting of *fully-extended*, orthogonal iCRSs, the proofs of the key properties which do not hold in the *non-fully-extended* case are centred around two intermediate results. Given a term s, these results are as follows:

- if a perpetual reduction starts from s, then s has no head normal form, and
- if a hypercollapsing reduction starts from s, then s is hypercollapsing.

The proofs can, resp., be found in [6] and [10].

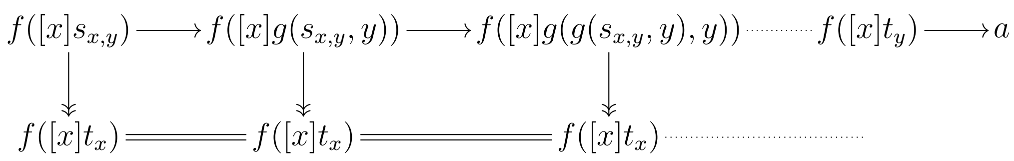

Figure 1. Preliminary diagram for the failure of the intermediate properties

The intermediate results do not follow in case non-fully-extended rules are allowed. To see this, consider the next two rules:

$$f([x]Z) \to a \tag{1}$$
$$g(Z, Z') \to Z \tag{2}$$

The first rule is non-fully-extended, as x does not occur as an argument of Z; the second is from the introduction.

Employing rules (1) and (2) and considering the terms $s_{x,y} = g(g(s_{x,y}, y), x)$ and $t_x = g(t_x, x)$, we can construct the diagram in Figure 1. In the diagram, $f([x]s_{x,y})$ reduces to both a and $f([x]t_x)$, with a irreducible and $f([x]t_x)$ a head normal form only reducible to itself. Hence, $f([x]s_{x,y})$, whose x is used in a non-occur check, reduces both to a term in which x does not occur and to a term from which x cannot be removed by further reduction.

To refute the properties from above, consider two additional rules:

$$a \to h([x]h(x)) \tag{3}$$
$$h([x]Z(x)) \to Z([x]Z(x)) \tag{4}$$

Rule (3), in combination with collapsing rule (4), ensures that a hypercollapsing and, hence, a perpetual reduction starts from a: The term $h([x]h(x))$ is a redex and reduces only to itself. Thus, $f([x]s_{x,y})$ now reduces both to a head normal form and has a hypercollapsing and, hence, a perpetual reduction starting from it, refuting both properties from above.

In the fully-extended case [10,6], the above properties are proved by showing that a hypercollapsing, resp., a perpetual reduction can be projected over a single (non-root) step. By adding the following rule, we can also refute these two projection properties in case non-fully-extended rules may occur:

$$k([x]Z(x)) \to Z(k[x]Z(x)) \tag{5}$$

Construct the diagram in Figure 2. As the diagram 'fits' to the left of the diagram in Figure 1 and as the left-hand side consists of a single (non-root) step, failure of the projection properties follows.

$$
\begin{array}{ccc}
f([x]k([z]g(g(z,y),x))) & \twoheadrightarrow & f([x]s_{x,y}) \\
\downarrow & & \downarrow\!\!\downarrow \\
f([x]k([z]g(z,x))) & \twoheadrightarrow & f([x]t_x)
\end{array}
$$

Figure 2. Diagram that 'fits' to the left of the diagram in Figure 1

4 Failure of Central Properties

We show failure of confluence modulo hypercollapsing subterms—confluence already fails in the infinitary first-order case [4,13], as is witnessed by the reductions $t_x \twoheadleftarrow s_{x,y} \twoheadrightarrow t_y$. We also show that the fair reduction strategy fails to be normalising and that the normal form property no longer holds.[2] We employ rules (1) and (2) from the previous section.

Confluence. Write $s \sim_{hc} t$ if t can be obtained from s by replacing (possibly infinitely many) hypercollapsing subterms in s by other hypercollapsing terms. An iCRS is confluent *modulo hypercollapsing subterms*, or $\sim_{hc}$, if for all $s' \twoheadleftarrow s \sim_{hc} t \twoheadrightarrow t'$ there exist terms s'' and t'' such that $s' \twoheadrightarrow s'' \sim_{hc} t'' \twoheadleftarrow t'$. While confluence modulo $\sim_{hc}$ holds for fully-extended, orthogonal iCRSs [9,10], it fails in case non-fully-extended rules are allowed: Consider t_x and t_y, which are hypercollapsing by rule (2). We have $f([x]t_x) \sim_{hc} f([x]t_y)$. However, $f([x]t_y) \to a$ while $f([x]t_x)$ reduces only to itself. As no hypercollapsing subterms occur in a and as $f([x]t_x)$ is a head normal form, we have $f([x]t_x) \not\sim_{hc} a$ and failure of confluence modulo $\sim_{hc}$ follows.

Holding against the above counterexample is the observation that by going from $f([x]t_x)$ to $f([x]t_y)$ we replace a subterm whose variables are all bound by one whose variables are all free. Hence, one might conjecture that it suffices to replace confluence modulo $\sim_{hc}$ by the weaker *confluence up to* $\sim_{hc}$, i.e. where for each $s' \twoheadleftarrow s = t \twoheadrightarrow t'$ we have $s' \twoheadrightarrow s'' \sim_{hc} t'' \twoheadleftarrow t'$. However, this is not the case. To see this, consider again the diagram in Figure 1. The term $f([x]s_{x,y})$ reduces both to $f([x]t_x)$ and a, and, as $f([x]t_x) \not\sim_{hc} a$, confluence up to $\sim_{hc}$ does not hold either.

Fair Normalisation. A reduction is *fair* if for every redex in some term along the reduction it holds finitely many steps later that either (1) a residual of the redex is contracted or that (2) no residual of the redex occurs. In case of fully-extended, orthogonal iCRSs, fair reductions are normalising by strongly convergent reductions [11].

---

[2] Compression is already shown to fail in [2], although the top reduction in Figure 1 also provides a counterexample.

To see that fair reductions are not necessarily normalising, or even strongly convergent, in case non-fully-extended rules occur, consider the following cyclic reduction, where we repeatedly contract the outermost redex (note the subscripts of s):

$$f([x]f([y]s_{x,y})) \rightleftarrows f([x]f([y]s_{y,x}))$$

As each step contracts the unique outermost redex, and as this redex is collapsing and does not create any new redexes, the reduction is fair. However, the term a, the unique normal form of both $f([x]f([y]s_{x,y}))$ and $f([x]f([y]s_{y,x}))$, is never obtained. Moreover. the reduction is not strongly convergent either.

Normal Form Property. Assuming $(\twoheadleftarrow\cdot\twoheadrightarrow)^*$ is the symmetric, transitive, reflexive closure of $\twoheadrightarrow$, the *normal form property* [13,4] states that $s\ (\twoheadleftarrow\cdot\twoheadrightarrow)^*\ t$ implies $s \twoheadrightarrow t$ in case t is a normal form. The property holds for fully-extended, orthogonal iCRSs [10]. If non-fully-extended rules are allowed, this is no longer the case. Consider again the diagram in Figure 1: The term $f([x]s_{x,y})$, which has the normal form a, reduces to $f([x]t_x)$, which only reduces to itself and not to a, while reduction to a is required for the normal form property to hold.

5 Conclusion and Further Research

The counterexamples in the current paper come about due to the interaction of non-fully-extended and collapsing rules, the latter of which already give rise to non-confluence [4,9,10]. Hence, it might be interesting to explore systems that do not allow for collapsing rules. However, such systems are of limited use as far as modelling is concerned: Both the β- and η-rule from λ-calculus and the `head` and `tail` operations on lists, which occur in functional programming, are collapsing. Other interesting research directions include investigating reduction modulo $\sim_{hc}$, which is even unexplored in the first-order infinitary setting, and re-obtaining confluence by means of introducing meaningless terms [5,7] in a setting with non-fully-extended rules.

Two properties which hold for fully-extended, orthogonal iCRSs have not been refuted in the current setting: unique normalisation (with respect to reduction) [10] and uniform normalisation [6]. We conjecture that these results do hold in case non-fully-extended rules may occur. Failure of unique normalisation would require a term to have at least two normal forms, which seems to be impossible; uniform normalisation is easily shown to rule out the occurrence of collapsing rules.

Acknowledgements

The author wishes to thank Jakob Grue Simonsen and the anonymous referees for their useful comments and remarks.

References

[1] M. Hanus, C. Prehofer, Higher-order narrowing with definitional trees, in: Proc. of the 7th Int. Conf. on Rewriting Techniques and Applications (RTA '96), vol. 1103 of LNCS, 1996.

[2] J. R. Kennaway, J. W. Klop, M. R. Sleep, F.-J. de Vries, Infinitary lambda calculus, TCS 175 (1) (1997) 93–125.

[3] R. Kennaway, F.-J. de Vries, Infinitary rewriting, in: Terese [13], Chapter 12, pp. 668–711.

[4] R. Kennaway, J. W. Klop, R. Sleep, F.-J. de Vries, Transfinite reductions in orthogonal term rewriting systems, I&C 119 (1) (1995) 18–38.

[5] R. Kennaway, V. van Oostrom, F.-J. de Vries, Meaningless terms in rewriting, JFLP 1999 (1).

[6] J. Ketema, On normalisation of infinitary combinatory reduction systems, in: Proc. 19th Int. Conf. on Rewriting Techniques and Applications (RTA 2008), vol. 5117 of LNCS, 2008.

[7] J. Ketema, Comparing Böhm-like trees, in: Proc. 20th Int. Conf. on Rewriting Techniques and Applications (RTA 2009), vol. 5595 of LNCS, 2009.

[8] J. Ketema, J. G. Simonsen, Infinitary combinatory reduction systems, in: Proc. 16th Int. Conf. on Rewriting Techniques and Applications (RTA 2005), vol. 3467 of LNCS, 2005.

[9] J. Ketema, J. G. Simonsen, On confluence of infinitary combinatory reduction systems, in: Proc. 12th Int. Conf. on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR 2005), vol. 3835 of LNAI, 2005.

[10] J. Ketema, J. G. Simonsen, Infinitary combinatory reduction systems: Confluence, LMCS 5 (4:3) (2009) 1–29.

[11] J. Ketema, J. G. Simonsen, Infinitary combinatory reduction systems: Normalising reduction strategies, LMCS 6 (1:7) (2010) 1–35.

[12] J. W. Klop, V. van Oostrom, F. van Raamsdonk, Combinatory reduction systems: introduction and survey, TCS 121 (1 & 2) (1993) 279–308.

[13] Terese (ed.), Term Rewriting Systems, Cambridge University Press, 2003.

[14] V. van Oostrom, Higher-order families, in: Proc. 7th Int. Conf. on Rewriting Techniques and Applications (RTA '96), vol. 1103 of LNCS, 1996.

[15] F. van Raamsdonk, Confluence and normalisation for higher-order rewriting, Ph.D. thesis, Vrije Universiteit, Amsterdam (1996).