Church-Rosser Properties of Normal Rewriting[*]

Jean-Pierre Jouannaud[1,2] and Jianqi Li[3,4]

1 **INRIA-LIAMA, Beijing, China**
2 **Software Chair, Tsinghua University, Beijing, China**
3 **School of Software, Tsinghua University, Beijing, China**
4 **Tsinghua National Laboratory for Information Science and Technology, Beijing, China**

Abstract

We prove a general purpose abstract Church-Rosser result that captures most existing such results that rely on termination of computations. This is achieved by studying *abstract normal rewriting* in a way that allows to incorporate positions at the abstract level. New concrete Church-Rosser results are obtained, in particular for higher-order rewriting at higher types.

1998 ACM Subject Classification F4.2. Grammars and Other Rewriting Systems

Keywords and phrases abstract normal rewriting, Church-Rosser property

Digital Object Identifier 10.4230/LIPIcs.CSL.2012.350

1 Introduction

Background. Rewrite rules have been used in mathematics and computer science for ages. Orienting an equation into a rewrite rule is most convenient when the obtained set of rules is terminating, since many other properties, such as "unique normal form", become then decidable. Orienting all equations at hand into rewrite rules is not always possible, forcing us to distinguish a subset of rules from the subset of remaining equations. Consider the set: $Inv : x + x^{-1} = 0, \quad Z : x + 0 = x, \quad A : (x + y) + z = x + (y + z), \quad C : x + y = y + x.$ Termination forbids orienting C, and orienting A contradicts termination in C-congruence classes: it becomes necessary to distinguish rules from equations. Having A, C, Z as equations allows for cheaper pattern-matching and unification than A, C alone, but ACZ-congruence classes are infinite, raising problems. A winning schema is to restrict computations to terms in Z normal form modulo AC [4, 11, 16]. Rewriting with Inv then operates modulo ACZ, but on terms in Z modulo AC normal form.

Goal. In this paper, we investigate rewriting with a set of rules R (Inv in our example) modulo a set itself made of a set of rules S (Z) and a set of equations E (AC). Another example is Nipkow's higher-order rewriting, for which R is the user's set of higher-order rules, S corresponds to β-reduction and η-expansions which are built, together with α-conversion (E), in the rewriting mechanism via higher-order pattern-matching. Higher-order rewriting is indeed our main target, and our interest is in checking its confluence under some termination assumption.

State of the art. Rewriting with rules only (when E and S are both empty) is called *plain rewriting*. Confluence of plain rewriting reduces, under a termination assumption, to the

[*] INRIA Project FORMES, Laboratory of Formal Methods, Institute of Software Theory and System, Tsinghua University. This work was supported by the Tsinghua National Laboratory for Information Science and Technology (TNList) Cross-discipline Foundation 2011-9, Chinese National 973 Plan grant 2010CB328003, the NSFC-ANR grant 60811130468 and NSFC grant 60903030.

joinability of its *critical pairs*, which are minimal divergent computations obtained by unifying lefthand sides of rules at non-variable subterms [14]. When rewriting terms with mixed sets of rules and equations, confluence must be replaced by the more general *Church-Rosser* property [10]. There are several approaches corresponding to different definitions of rewriting in presence of equations. In the first three, S is empty. Huet uses plain rewriting with R and joinability modulo E, but must assume strong linearity conditions on E and R [9]. Lankford uses plain rewriting with R in equivalence classes of terms modulo E [15]. His *class rewriting* may require searching the entire class of a term to be rewritten. The de facto standard introduced by Peterson and Stickel, *rewriting modulo*, uses instead plain rewriting with all possible E-variants of the instances of rules in R, which is implemented via E-pattern matching [21]. All three approaches require finiteness of E-equivalence classes [10], see also [2]. To remove this assumption, Marché defined *normalized rewriting*, a complex schema where the rules in S are confluent modulo E in Stickel's sense and a term is first normalized with S modulo E before being rewritten with R modulo E [16]. Normalized rewriting assumes termination of $R \cup S$ in E-equivalence classes. Nipkow uses a subtle variation of the latter requiring termination of R modulo $E \cup S$, in which simply typed higher-order terms *in normal form* for S modulo E are rewritten modulo $S \cup E$ (using higher-order pattern-matching) with some set of higher-order rules which lefthand sides are patterns *à la* Miller [17].

In all these approaches, the Church-Rosser property is reduced, via a termination assumption, to the joinability modulo E of some critical pairs. This reduction is doable provided rewriting a term does not change its structure before it is pattern-matched. This is the case of all definitions but that of class rewriting and of normalized rewriting. In the latter case, the analysis of divergent computations leads to a notion of critical pair which is quite complex. On the other hand, Nipkow's variant, which we call *normal rewriting*, has not lead yet to a general Church-Rosser test.

Very early on, the rewriting community has developed an abstract approach to the analysis of the confluence and Church-Rosser properties [23], or to similar results such as the finite development theorem in lambda calculus [8, 22]. This trend has been very successful for orthogonal systems and extensions thereof, but has not yet delivered all its promises for the case of terminating systems for which the presence of critical pairs requires a slightly different analysis. A tentative to capture both cases within a unique framework, decreasing diagrams, has been carried out by van Oostrom [20]. If this work has been very successful at the abstract level of relations, it has not yet bared fruits in the more difficult case of concrete relations over terms.

Contributions. Our main contribution is a careful investigation of an abstract rewriting relation, *normal rewriting*, with a set of rules R modulo a pair (S, E), which set of rules S is itself convergent modulo the set of equations E. Normal rewriting is then defined as a compositional paradigm: normalization in the $S \cup E$-structure is *modulo* E, while normalization in the $(R \cup (S \cup E))$-structure is modulo $S \cup E$ on S modulo E normal forms. We then reduce the abstract Church-Rosser property to properties of *abstract critical rewriting patterns*. Our abstract treatment departs from the usual practice by introducing a setting of ternary relations on an abstract set of terms and an abstract set of positions, which we call *abstract positional rewriting*. This setting allows us to develop our notion of abstract normal rewriting, and then to study the reduction from its Church-Rosser property to the critical rewriting patterns quite smoothly, therefore solving the aforementioned problem.

Our second contribution, an application of the above results, is a careful investigation of the Church-Rosser properties of first-order normal rewriting first, then of various variants of Nipkow's higher-order normal rewriting. These applications are direct reductions from abstract critical patterns to concrete critical pairs, which exploit the term structure explicitly. A major strength of our result is that it allows to capture the existing notions of rewriting in both the first-order and higher-order cases. While its application to the first-order case yields limited new results, it allows us in the higher-order case to overcome all typing restrictions imposed in Nipkow's work.

While it may seem that rewriting is a subject beaten to death, recent work has shown that confluence of normal rewriting is indeed at the heart of important problems such as [1], hence making the present contributions very timely. An early attempt appeared in [12].

Surveys are [7, 23, 22] for first and higher-order rewriting and [5] for typed lambda calculus.

Organization. Section 2 investigates the Church-Rosser properties of abstract normal rewriting. First-order rewriting is developed in Section 3 which ends with a study of our introductory example. After a brief introduction of simply-typed lambda-calculi, higher-order rewriting at simple types is studied in section 4.1 and at higher-types in Section 4.2. Various definitions of formal derivation serve illustrating the results. Concluding remarks come in Section 5.

2 Normal rewriting

We introduce the notion of normal rewriting on an abstract set, investigating then its Church-Rosser properties. Our treatment differs from similar attempts by introducing abstract positions from the start. This allows us to carry out the investigation of the abstract Church-Rosser property much further, and reduce it to the joinability of abstract critical patterns via clean, technically simpler proofs than in the current literature. The ability to analyze the Church-Rosser property at the abstract level up to the analysis of critical pairs is the key to the obtention of our main result, Theorem 2.5, which proof is quite delicate.

2.1 Abstract positional rewriting

2.1.1 Abstract terms and positions

In the entire Section 2, we assume given two abstract sets:

- $\mathcal{T}$ which elements are called *terms* ;
- $\mathcal{P}$ which elements are called *positions*, equipped with a partial order $>_{\mathcal{P}}$ and a minimum Λ.

A *domain* P_p is any set of positions $p' \geq_{\mathcal{P}} p$ such that $p' \in P_p$ and $p' \geq_{\mathcal{P}} q \geq_{\mathcal{P}} p$ implies $q \in P_p$. A domain is meant to be the set of non-variable positions of some lefthand side of rule in a term.

Lexicography: we shall use the letters s, t, u, v, w for terms and p, q for positions, the notations P_p and Q_q for *domains*, and the notation $\mathcal{D}_{\mathcal{P}}$ for the set of domains over $\mathcal{P}$. We write $p\#q$ for incomparable positions p, q, and $q >_{\mathcal{P}} P_p$ for $q \geq_{\mathcal{P}} p$ and $\forall p' \in P_p \,.\, p' \not\geq_{\mathcal{P}} q$. We will freely use the following key property of domains, which first 3 cases are called respectively *"disjoint case"*, *"critical case"* and *"ancestor case"* in the litterature:

$$\forall p \in \mathcal{P}\ \forall P_p \in \mathcal{P}.(q\#p \vee q \in P_p \vee q >_{\mathcal{P}} P_p \vee p >_{\mathcal{P}} q).$$

2.1.2 Relations

We use the notation $u\xrightarrow{P_p}v$ with $u, v \in \mathcal{T}$, $P_p \in \mathcal{D}_{\mathcal{P}}$ for an arbitrary ternary relation in $\mathcal{T}\times\mathcal{D}_{\mathcal{P}}\times\mathcal{T}$. We may omit any of u, v, P_p, in which case they are existentially quantified. We also write for short $u\xrightarrow{p}v$, where p is understood as the minimum of some domain P_p, or $u\xrightarrow{p\in P}v$ to indicate the property P that the position p should satisfy, or even $u\xrightarrow{P}v$, with the same meaning. In practice, P will usually be the property $(\geq_{\mathcal{P}}\ q)$ for some given q, characterizing the set of positions $\{p \mid p\geq_{\mathcal{P}}\ q\}$.

The relation $\longrightarrow$ is *reflexive* if $\forall u \in \mathcal{T}\,.\,u\longrightarrow u$, *symmetric* if for all s, t, p, $s\xrightarrow{p}t$ iff $t\xrightarrow{p}s$, and *transitive* if for all u, v, w s.t. $u\longrightarrow v\longrightarrow w$ then $u\longrightarrow w$. Given $\longrightarrow$, we write $\longleftarrow$ for its inverse, $\longleftrightarrow$ for its symmetric closure, $\xrightarrow{+}$ for its transitive closure and $\xrightarrow{*}$ for its reflexive, transitive closure (domains become lists thereof in the latter two closures).

The term t is a *successor below* $p \in \mathcal{P}$ of s for $\longrightarrow$ if $s\xrightarrow{\geq_{\mathcal{P}}\ p}t$, and s is in *normal form below* p if it has no successor below p. We denote by $s{\downarrow}^p$ the term in normal form below p such that $s\xrightarrow{(\geq_{\mathcal{P}}\ p)^*}s{\downarrow}^p$. We omit the mention *below* p and write $s{\downarrow}$ whenever $p = \Lambda$. A term s is *strongly normalizing* below p for $\longrightarrow$ iff it is in normal form below p, or if otherwise all its successors are themselves strongly normalizing below p. The relation $\longrightarrow$ is *terminating* if all terms are strongly normalizing below Λ.

2.1.3 Rewriting modulo

▶ **Definition 2.1** (Rewriting modulo). Given two relations $\xrightarrow{p}_X$ and $\underset{Z}{\overset{q}{\longleftrightarrow}}$ (assumed symmetric) on $\mathcal{T} \times \mathcal{D}_{\mathcal{P}} \times \mathcal{T}$, *rewriting with* X *modulo* Z *at* p is defined as:

$$\xrightarrow{p}_{X_Z} \quad := \quad \underset{Z}{\overset{(\geq_{\mathcal{P}}\ p)^*}{\longleftrightarrow}}\ \xrightarrow{p}_X$$

Z and *modulo* Z are omitted if $Z = \emptyset$. The symmetric closure of Z should be understood in case Z is not symmetric.

The beauty of rewriting modulo a theory lies in the assumption that equality steps can only occur below the rewriting position: at the term level, rules can be fired by pattern matching lefthand sides modulo the theory, avoiding searching the equivalence class of the term to be rewritten.

Rewriting modulo is assumed to satisfy the *commutation* (*) and *joinability* (**) properties:

$$(*) \qquad {}_X\xleftarrow{p}\ \xrightarrow{q}_Y \subseteq \xrightarrow{p}_Y\ {}_X\xleftarrow{q} \qquad \text{if} \quad p\#q$$

$$(**) \quad {}_X\xleftarrow{P_p}\ \xrightarrow{q}_Y \subseteq \xrightarrow{(>_{\mathcal{P}}\ p)^*}{}_Y\ {}_X\xleftarrow{q}\ \ {}_Y\xleftarrow{(>_{\mathcal{P}}\ P_p)^*} \qquad \text{if} \quad q>_{\mathcal{P}} P_p$$

Note that (*) is a particular case of (**), apart from the condition which could be generalized. We prefer to distinguish these two axioms because they lead to different calculations in the proof of our main result. Note also the absence of a "monotonicity" axiom allowing to move rewrites up, which is achieved by changing the position incorporated to the rewrite.

2.1.4 Normal rewriting

▶ **Definition 2.2.** A *normal abstract rewriting system* (NARS) is a tuple $(\mathcal{T}, \mathcal{P}, R, S, E)$, where $\mathcal{T}$ and $\mathcal{P}$ are (possibly omitted) abstract sets of terms and positions, while $\longrightarrow_R$, $\longrightarrow_S$

and $\xleftrightarrow[E]{}$ are ternary relations on $\mathcal{T} \times \mathcal{D}_{\mathcal{P}} \times \mathcal{T}$ called respectively *rewriting* R, *simplification* S and *equality* E, the latter being supposed symmetric, such that:

(i) the relation S_E, called E-*simplification*, is *Church-Rosser modulo* E *below* any position p:

$$s \xleftrightarrow[S \cup E]{(\geq_{\mathcal{P}}\ p)^*} t \quad \text{iff} \quad s \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}{}_{S_E} \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} {}_{S_E}\xleftarrow{(\geq_{\mathcal{P}}\ p)^*} t$$

(ii) $\longrightarrow_{R_{SE}} \cup \longrightarrow_{S_E}$ is E-*terminating*: the relation $s \succ t$ iff $s =_E (\longrightarrow_{R_{SE}} \cup \longrightarrow_{S_E}) =_E t$ is well-founded, where R_{SE} stands for *rewriting* with R *modulo* $S \cup E$

(iii) operating on terms in $\downarrow^p_{S_E}$ normal form, *normal rewriting at* q *below* p *with* R *modulo* (S, E) is defined as $\quad s \xrightarrow{p}_{R_{S_E \downarrow}} t \ = \ s \xrightarrow{q \geq_{\mathcal{P}}\ p}_{R_{SE}} u\downarrow^p_{S_E} t.$

The main reason for orienting simplifiers is indeed to bypass two fundamental assumptions of rewriting modulo: *termination* of $\longrightarrow_R$ in *finite* $S \cup E$-equivalence classes. The finiteness assumption will disappear, while the termination assumption will be weakened.

Normal rewriting maintains S_E normal forms below p, and satisfies (*, **). Nipkow's definition assumes $p = \Lambda$. Our definition is relative to a given position p, as are all our rewriting notions. This definition is actually flexible. We might have chosen to operate on terms in$\downarrow^q_{S_E}$-normal form or to normalize the result up to position Λ. These changes would not impact our result for which p is Λ.

Notations. We use, possibly omitting the upper-index p: $\quad s \xrightarrow{\downarrow} t$ for $s \xrightarrow{*} t$ with $t = t\downarrow$, $s\downarrow^p$ for $s\downarrow^p_{S_E}$, $\quad s\Downarrow^p$ for $(s\downarrow^p_{S_E})\downarrow^p_{R_{S_E\downarrow}}$, $\quad =_E$ for the equivalence $\xleftrightarrow[E]{*}$, $\xleftrightarrow[SE]{}$ (SE) for $\xleftrightarrow[S]{} \cup \xleftrightarrow[E]{}$, and $\xleftrightarrow[RSE]{}$ (RSE) for $\xleftrightarrow[R]{} \cup \xleftrightarrow[S]{} \cup \xleftrightarrow[E]{}$.

2.2 Church-Rosser properties of NARS

Our goal here is to reduce the Church-Rosser property of a terminating NARS to local properties of the rewrite relations involved that can be checked for concrete rewrite relations on terms. After introducing key notations, we recall some Church-Rosser notions and introduce our local properties which correspond *exactly* to critical patterns.

▶ **Definition 2.3.** We define:
convertibility of a pair (s, t) below p as $s \xleftrightarrow[RSE]{(\geq_{\mathcal{P}}\ p)^*} t$;
normal joinability of a convertible pair (s, t) below p as $s\Downarrow^p \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ )^*} t\Downarrow^p$
normal Church-Rosser as the normal-joinability of all convertible pairs ;
local peaks (resp. *cliffs*) as triples (s, u, t) s.t. $s \xleftarrow{p} u$ (resp. $s \xleftrightarrow[E]{p} u$) and $u \xrightarrow{q} t$;
joinability below p of a triple (s,u,t) or a pair (s,t) as $s \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}{}_{S_E \cup R_{SE}} \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} {}_{S_E \cup R_{SE}}\xleftarrow{(\geq_{\mathcal{P}}\ p)^*} t.$

▶ **Definition 2.4** (Critical local peaks and cliffs).

(i) *critical rewrite peak* $\quad v\ {}_R\xleftarrow{P_p} u \xrightarrow{q}_{R_{SE}} w \quad$ s.t. $q \in P_p$ and $u = u\downarrow^p_{S_E}$
(ii) *critical rewrite cliff* $\quad s \xleftrightarrow[E]{p} u \xrightarrow{q}_{R_{SE}} t \quad$ s.t. $q \in P_p \setminus \{p\}$
(iii) *critical simplification peak* $\quad v\ {}_R\xleftarrow{P_p} u \xrightarrow{q}_{S_E} w \quad$ s.t. $q \in P_p$
(iv) *critical simplification cliff* $\quad v\ {}_S\xleftarrow{p} u \xrightarrow{q}_{R_{SE}} w \quad$ s.t. $q \in P_p \setminus \{p\}$ and $u = u\downarrow^q_{S_E}$

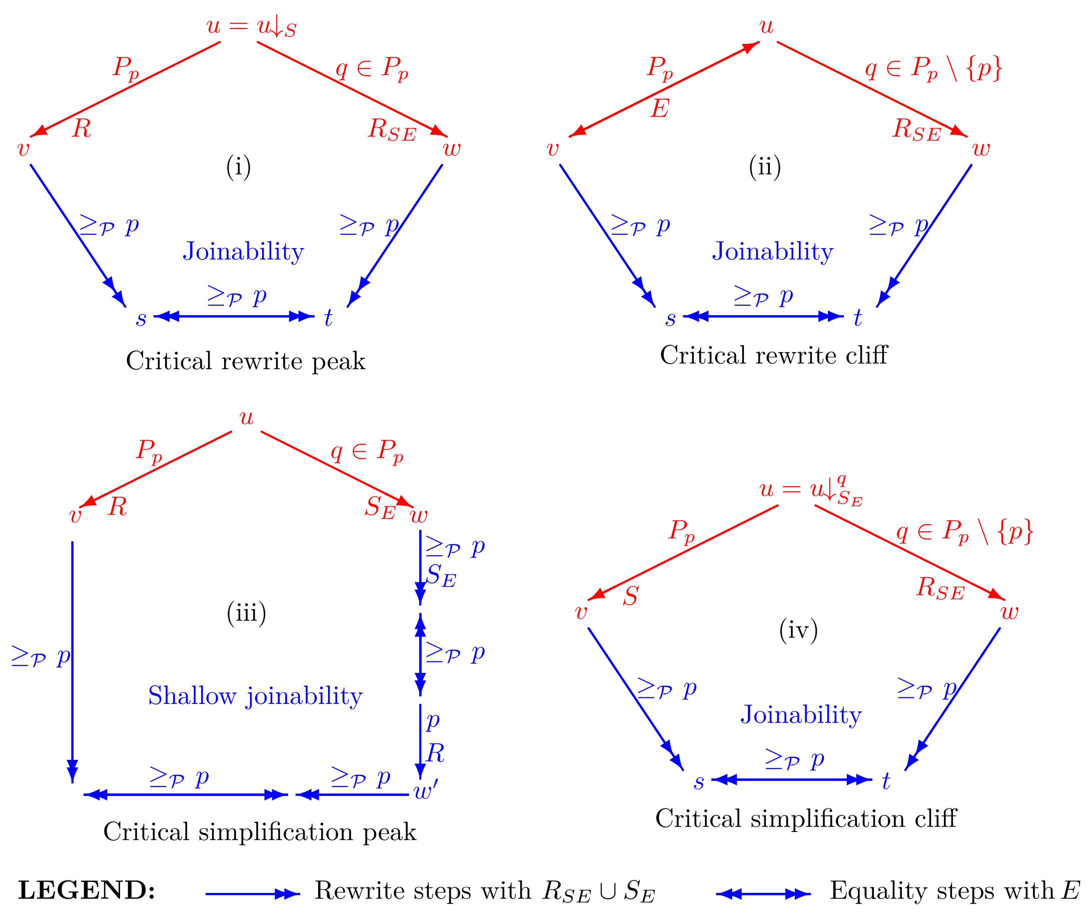


Figure 1 Abstract critical peaks.

Unlike standard practice, our local properties are *critical* in that they specialize as the usual notions of critical peaks at the concrete level. Criticality follows from two observations: (i) each peak must satisfy the condition $q \in P_p$ (or $q \in P_p \setminus \{p\}$), which implies the existence of an overlap; (ii) all local properties use a plain step from u at p, this is crucial to compute a critical pair by equating two different calculations of $u|_q$, requiring the absence of equational steps above q.

Our figures also show the properties expected from the critical peaks: joinability below p for all except critical simplification peaks, for which *shallow joinability* is required.

We can now state and prove our key abstract Church-Rosser result, without help of any intermediate step involving non-local peaks corresponding to local confluence, local coherence and the like: despite the complex hierarchical structure of a NARS, our abstract approach allows us to reduce directly the Church-Rosser property to the joinability of the critical peaks introduced above.

▶ **Theorem 2.5.** *A NARS is normal Church-Rosser if its critical (i) rewrite peaks, (ii) rewrite cliff, (iii) simplification cliffs are joinable, and its (iv) critical simplification peaks are shallow joinable.*

Proof. By definition of rewriting modulo, $\xleftrightarrow[RSE]{*} = \xleftrightarrow[R_{SE} \cup S_E \cup E]{*}$. The proof is by induction on conversions $\xleftrightarrow[R_{SE} \cup S_E \cup E]{*}$. Conversions are interpreted by multisets which elements are pairs of terms (u, v) written as $u\,v$, and are therefore compared in the well-founded order

$\succ\!\!\succ := ((\succ)_{lex})_{mul}$. Each step in a conversion contributes with one or two pairs: a step $s \longrightarrow_{R_{SE}} t$ with $s\,t$; a step $s \longrightarrow_{S_E} t$ with $t\,s$; a step $s \xleftrightarrow[E]{} t$ with both $s\,t$ and $t\,s$.

By definition, the shape of a normal joinable conversion $s \xleftrightarrow[R_{SE}\cup S_E\cup E]{*} t$ must be of the form $s \xrightarrow{\downarrow}_{S_E} (\longrightarrow_{R_{SE}} \xrightarrow{\downarrow}_{S_E})^* \xleftrightarrow[E]{*} ({}_{R_{SE}}\!\longleftarrow {}_{S_E}\!\xleftarrow{\downarrow})^* {}_{S_E}\!\xleftarrow{\downarrow} t$. It follows that conversions which are not already normal joinable must contain one of the six (up to symmetry) following patterns: $u \xrightarrow{p}_{R_{SE}} v$ with $u \neq u{\downarrow^p}$, $v\ {}_{R_{SE}}\!\xleftarrow{p} u \xrightarrow{q}_{R_{SE}} w$ with $u = u{\downarrow^p}$ and $u = u{\downarrow^q}$, $v \xleftrightarrow[E]{} u \xrightarrow{p}_{R_{SE}} w$ with $u = u{\downarrow^p}$, $v\ {}_{S_E}\!\longleftarrow u \xrightarrow{p}_{R_{SE}} w$ with $u = u{\downarrow^p}$, $v\ {}_{S_E}\!\longleftarrow u \longrightarrow_{S_E} w$, and $v \xleftrightarrow[E]{} u \longrightarrow_{S_E} w$. In each case, we provide a smaller conversion for (v, w) yielding a smaller conversion for (s, t), hence allowing us to conclude by induction. We are indeed rewriting conversions in the style of [3].

1. $v\ {}_{S_E}\!\longleftarrow u \longrightarrow_{S_E} w$, a local peak interpreted by $\{u\,v, w\,u\}$. By definition of a NARS, we have $v \xrightarrow{*}_{S_E} \xleftrightarrow[E]{*} {}_{S_E}\!\xleftarrow{*} w$, which interpretation contains pairs all (strictly) smaller than $u\,v$ (or $w\,u$).

2. $v \xleftrightarrow[E]{} u \longrightarrow_{S_E} w$, a cliff interpreted by $\{v\,u, u\,v, w\,u\}$. By definition of a NARS, we have $v \longrightarrow_{S_E} v' \xrightarrow{*}_{S_E} \xleftrightarrow[E]{*} w'\ {}_{S_E}\!\xleftarrow{*} w$, which interpretation contains pairs all smaller than $v\,u$.

3. $u \xrightarrow{p}_{R_{SE}} v$ with $u \neq u{\downarrow^p}$, a step which interpretation is $\{u\,v\}$. By definition of normal form below p, $u \xrightarrow{q}_{S_E} w$ (hence $u \succ w$) for some w and $q \geq_{\mathcal{P}} p$. By definition of rewriting modulo, $w \longrightarrow_{R_{SE}} v$. The resulting conversion is interpreted by the smaller multiset $\{w\,u, w\,v\}$.

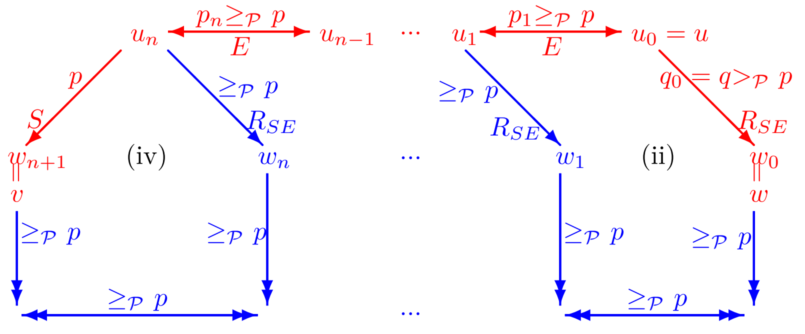
4. $v \xleftrightarrow[E]{p} u \xrightarrow{Q_q}_{R_{SE}} w$, a cliff interpreted by $\{v\,u, u\,v, u\,w\}$. We conclude by (*) if $p \# q$, definition of R_{SE} if $p \geq_{\mathcal{P}} q$, (**) if $p >_{\mathcal{P}} Q_q$ and assumption (ii) if $p \in Q_q \setminus \{q\}$. In all cases, the obtained proof is interpreted by pairs which are strictly smaller than $v\,u$ or $u\,w$.

5. $v\ {}_{S_E}\!\xleftarrow{P_p} u \xrightarrow{q}_{R_{SE}} w$, a peak interpreted by $\{v\,u, u\,w\}$. Thanks to the first case, we can assume that $u = u{\downarrow^q}$. There are therefore three possible cases:

– $p \# q$. Then $v \xrightarrow{q}_{R_{SE}} t\ {}_{S_E}\!\xleftarrow{p} w$ by (*), interpreted by $\{v\,t, t\,w\}$ smaller than $\{v\,u, u\,w\}$.

– $q >_{\mathcal{P}} P_p$. By (**), we get the (smaller) joinability proof $v \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}_{R_{SE}} v'\ {}_{S_E}\!\xleftarrow{p} w'\ {}_{R_{SE}}\!\xleftarrow{(\geq_{\mathcal{P}}\ p)^*} w$.

– $q \in P_p \setminus \{p\}$. By definition of rewriting modulo, $v\ {}_{S}\!\xleftarrow{p} u_n \xleftrightarrow[E]{p_n} \cdots \xleftrightarrow[E]{p_1} u_0 = u$, with $\forall i \in [1..n]\ p_i \geq_{\mathcal{P}} p$. Note that all u_i are in S_E-normal form below q since this is true of u. We now show the existence of terms w_i such that $u_i \xrightarrow{q_i \geq_{\mathcal{P}}\ p}_{R_{SE}} w_i$, hence $u \succ w_i$ (requiring here an order on E-equivalence classes), and $\forall i \in [1..n]$ the pair (w_{i-1}, w_i) is joinable with steps which interpretation is made of pairs smaller than $\{u\,w\}$. Letting $w_0 = w$ and $q = q_0$, we use an induction on i: the case $i = 0$ is by assumption. Assuming the property up to $i - 1$, we proceed as follows: if $q_i \# p_{i+1}$, by (*) ; if $p_{i+1} \geq_{\mathcal{P}} q_i$, by definition of rewriting modulo ; if $q_i >_{\mathcal{P}} P_{p_{i+1}}$, by (**) ; and if $q_i \in P_{p_{i+1}}$, by assumption (ii). We close the diagram by definition of rewriting modulo if $q_n = p$, by assumption (iv) applied to the peak $v\ {}_{S}\!\xleftarrow{p} u_n \xrightarrow{q_n}_{R_{SE}} w_n$ if $q_n \in P_p$ (see the coming picture) and by property (**) if $q_n >_{\mathcal{P}} P_p$, yielding a smaller conversion each time.

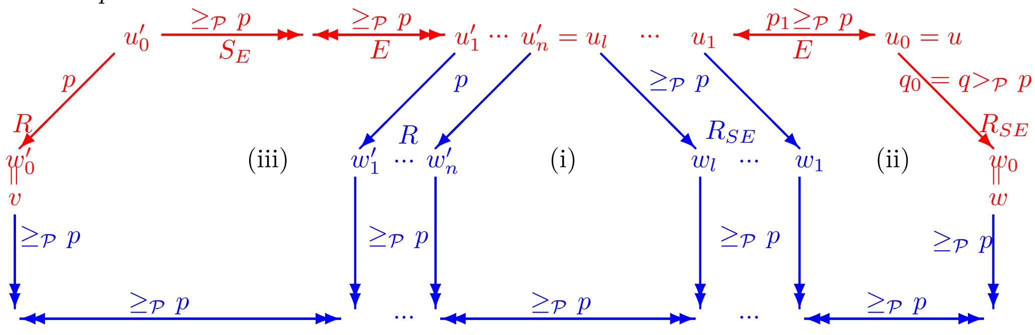


6. $v\ {}_{R_{SE}}\!\xleftarrow{P_p} u \xrightarrow{q}_{R_{SE}} w$. Thanks to the first case, we can assume wlog that u is in normal form below p and q, which comparison yields three cases up to symmetry:
– $p\#q$. Then $v \xrightarrow{q}_{R_{SE}} u'\ {}_{R_{SE}}\!\xleftarrow{p} w$ by (*), yielding a smaller conversion.
– $q >_{\mathcal{P}} P_p$. Property (**) yields a smaller conversion.
– $q \in P_p$. Then $v\ {}_{R}\!\xleftarrow{p} u'_0 \xleftrightarrow[SE]{(\geq_{\mathcal{P}}\ p)^*} u \xrightarrow{q}_{R_{SE}} w$ by definition of rewriting modulo, and u is in S_E-normal form below p. As shown on the picture below, we proceed in three steps.

First: we move $u'_0 \xrightarrow{p}_R$ from u'_0 to some u'_n in S_E-normal form s.t. $u'_n \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} u$ and $\forall i \in [1..n]$, $u'_{i-1} \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}_{S_E} \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} u'_i \xrightarrow{p}_R w'_i$ and the pair (w'_{i-1}, w'_i) is joinable. Hence $u \longrightarrow_{R_{SE}} w'_i$ and therefore $u \succ w'_i$. The proof of the claim is by induction on i. If $u'_0 = u'_0{\downarrow}_{S_E}$, we are done with $n = 0$ and $w'_0 = w$. Otherwise $u'_0 \xrightarrow{\geq_{\mathcal{P}}\ p}_S u'_1$. By assumption (iii) (case shown on the picture) or property (**), $u'_0 \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}_{S_E} \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} u'_1 \xrightarrow{p}_R w'_1$ and the pair (w'_0, w'_1) is joinable. S_E being Church-Rosser below p, $u'_1 \xrightarrow{(\geq_{\mathcal{P}}\ p)^*}_{S_E} \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ p)^*} u$. Induction hypothesis applied to u'_1 concludes.

Second: let $u = u_0$, $\forall i \in [1..l]$, $u_{i-1} \xleftrightarrow[E]{\geq_{\mathcal{P}}\ p} u_i$ and $u_l = u'_n$. We proceed moving $u_0 \xrightarrow{\geq p}_{R_{SE}}$ from u_0 to u_l showing the existence of terms w_i such that: $\forall i \in [0..l]$, $u_i \xrightarrow{q_i \geq_{\mathcal{P}}\ p}_{R_{SE}} w_i$ and the pair (w_{i-1}, w_i) is joinable below p. This is done by induction on i. Case $i = 0$ is clear. The step case uses: if $p_1 \# q_0$, assumption (*); if $p_1 \geq_{\mathcal{P}} q_0$, the definition of rewriting modulo, hence $w_1 = w_0$; if $q_0 >_{\mathcal{P}} p_1$, property (**) or assumption (ii) (the case shown on the picture). For all $i \in [0..l]$, the property $u \succ w_i$ follows from the fact that $u =_E u_i \longrightarrow_{S_E \cup R_{SE}} w_i$.

Third: we close the obtained local peak $w'_n\ {}_R\!\xleftarrow{p} u'_n = u_l \xrightarrow{q_l}_{R_{SE}} w_l$ thanks to property (**) or assumption (i) (case on the picture), using the fact that $u'_n = u_l$ is in S_E-normal form below p.



All steps in the resulting conversion $w'_0 \ldots w'_n \ldots w_l \ldots w_0$ are interpreted by pairs which we have shown to be all strictly smaller than $u\,w$, hence we are done. ◀

We are ready for applying our main result to a first-order, and then higher-order, concrete setting.

3 First-order rewriting systems

There are many versions of first-order rewriting, all covered by normal rewriting except class and normalized rewriting as discussed in introduction.

3.1 Terms and rules.

We denote by $\mathcal{T}(\mathcal{F}, \mathcal{X})$ the free algebra of *terms* generated from a signature $\mathcal{F}$ of function symbols and a denumerable set $\mathcal{X}$ of variables. We assume the usual definitions of terms, positions, and substitutions [7, 23], adopting notations from the former. We use $\mathcal{V}ar(t)$ for the set of variables of the term t, $\mathcal{P}os(t)$ for the set of positions in t, and $\mathcal{FP}os(t)$ for its set of non-variable positions, and $\cdot$ for concatenation of positions. The *subterm* of t at position p is denoted by $t|_p$, and we write $t[u]_p$ for the result of replacing $t|_p$ at position p in t by u. Positions are compared in the prefix ordering. Substitutions are homomorphic extensions of a map from variables to terms, to a map from terms to terms. t is an instance of s by the substitution σ if $t = s\sigma$, using postfix notation for the substitution operation also called *instantiation.* Computing σ is called *(plain) pattern matching.* Substitution τ *subsumes* substitution σ if $\sigma = \tau\gamma$ for some substitution γ. Two terms s, t are *unifiable* if $s\sigma = t\sigma$, and σ is called a (unique up to renaming of variables) most general (plain) unifier (mgu for short) when it is minimal wrt the (well-founded) subsumption partial order.

A rewrite system is a set of pairs called rewrite rules, written as $R = \{l_i \to r_i\}_i$, where l_i is a non-variable term called its *lefthand side* and r_i is a term called its *righthand side* such that $\mathcal{V}ar(r_i) \subseteq \mathcal{V}ar(l_i)$. A term u (*plain*) *rewrites* to a term v with the rule $l_i \to r_i$ at position $p \in \mathcal{P}os(u)$, if $u|_p = l_i\sigma$ for some substitution σ and $v = [r_i\sigma]_p$. $l_i\sigma$ is called a *redex* and $r_i\sigma$ its *reduct.* We write $u \xrightarrow{p\cdot\mathcal{FP}os(l_i)}_{l_i \to r_i} v$ or $u \xrightarrow{p}_R v$, or simply $u \xrightarrow{p} v$.

A set of equations E is a symmetric rewrite system, in which case rewriting with E at position p is written $s \overset{p}{\underset{E}{\longleftrightarrow}} t$. The conversion relation $=_E$ is called the equational theory of E. t is an E-instance of s with the substitution σ if $t =_E s\sigma$. Computing σ is called *E-pattern matching.* A substitution σ is an *E-unifier* of the terms s, t if $s\sigma =_E t\sigma$. We are interested in theories, like AC, having a finite complete set of unifiers $CSU(s, t)$ for an arbitrary pair (s, t) of terms: any unifier of $s = t$ is then an E-instance of a unifier in $CSU(s, t)$.

3.2 Plain rewriting [14]

Plain rewriting corresponds to empty sets S and E. Plain rewriting satisfies the properties (*,**). The termination assumption of the NARS implies termination of the relation $\longrightarrow_R$. Then, the Church-Rosser property of R reduces to the joinability of local rewrite peaks $s\ {}_{l\to r\in R}\xleftarrow{p} u \xrightarrow{q}_R t$ with $q \in \mathcal{FP}os(l)$.

▶ **Definition 3.1.** Given two rules $g \to d$ and $l \to r$ in R s.t. $\mathcal{V}ar(g) \cap \mathcal{V}ar(l) = \emptyset$, and a position $p \in \mathcal{FD}om(g)$ such that l and $g|_p$ unify with most general unifier σ, then $\langle d\sigma, (g[r]_p)\sigma\rangle$ is called a *plain critical pair* of $l \to r$ onto $g \to d$ at p, of which $l\sigma$ is the *overlap.*

The proof of the Knuth and Bendix theorem now reduces to the sole classical critical pair case:

▶ **Theorem 3.2** (Knuth and Bendix). *Assume R is terminating. Then R is Church-Rosser iff its plain critical pairs are joinable with $\longrightarrow_R$.*

3.3 Rewriting modulo [21, 10]

We consider here the case where the set R is Church-Rosser modulo a theory E such as associativity and commutativity, S being empty. Rewriting uses then pattern matching modulo E. Our assumption that $=_E \longrightarrow_{R_E} =_E$ is terminating is called E-termination of R. Again, rewriting modulo enjoys the properties (*,**), which is not true of plain rewriting in E-congruence classes of terms.

We need to define critical pairs modulo:

▶ **Definition 3.3.** Given two rules $g \to d$ and $l \to r$ in R s.t. $\mathcal{V}ar(g) \cap \mathcal{V}ar(l) = \emptyset$, and a position $p \in \mathcal{FD}om(g)$ s.t. l and $g|_p$ unify with a complete set of most general unifiers Σ, then $\{\langle d\sigma, (g[r]_p)\sigma\rangle \mid \sigma \in \Sigma\}$ is called a complete set of *E-critical pairs* of $l \to r$ onto $g \to d$ at p, of which $l\sigma$ is the *E-overlap*.

We then recall the notion of extension introduced by Peterson and Stickel in the AC-case, and by Jouannaud and Kirchner in the general case:

▶ **Definition 3.4.** Given a rule $g \to d \in E$ and a rule $l \to r \in R$ such that $\mathcal{V}ar(g) \cap \mathcal{V}ar(l) = \emptyset$, and a position $p \in \mathcal{FD}om(g) \setminus \{\Lambda\}$ such that l unifies with $g|_p$ modulo E, then the rule $g[l]_p \to g[r]_p$ is called an *E-extension* of R.

▶ **Theorem 3.5** (Jouannaud and Kirchner). *Assume R is E-terminating and closed under E-extensions. Then R is Church-Rosser modulo E provided all its E-critical pairs are joinable.*

Proof. We omit the proof of joinability of critical rewrite peaks under the assumption that E-critical pairs are joinable, and concentrate on the critical rewrite cliffs. Let $s \underset{E}{\overset{P_p}{\longleftrightarrow}} u \xrightarrow{q}_{R_{SE}} t$ with $g = d \in E$, $P_p = \mathcal{FP}os(g)$ and $l \to r \in R$. By monotonicity of rewriting, we can assume $p = \Lambda$ and $q \in \mathcal{FP}os(g) \setminus \{\Lambda\}$. Then $s = d\sigma$ and $t = g\sigma$ while $t|_q = l\sigma$ and $u = l[d\sigma]_p$. We get $(g|_q)\sigma =_E l\sigma$, hence $g[l] \to g[r] \in R$ by closure assumption. Since $s \underset{E}{\longleftrightarrow} (g[l]_q)\sigma$, then $s \longrightarrow_{R_E} t = (g[r]_p)\sigma$. ◀

As shown in [21], extensions are finitely many for AC, and more generally when E is a set of permutative axioms, since extensions of extensions are then useless.

As a simple illustrating example, let $E = AC$ and $R = \{x + 0 \to x\}$. E-termination is obvious since AC-equivalence classes are size-preserving while the rule is size decreasing. There are no E-critical pairs since $x + 0$ does not E-unify with 0. Finally, R happens to be closed under extensions by pure luck: since $(x + 0) + y =_{AC} (x + y) + 0$, the extension $(x + 0) + y \to x + y$ is indeed an AC-instance of the original rule.

3.4 Normal rewriting

We now come to the general case, where R, S, E are sets of rules and equations satisfying our termination assumption. Note that our version of normal rewriting below p satisfies our assumptions (*,**), which is not the case of Nipkow's variant for which terms are normalized above p, destroying both. We actually only need that R_{SE}, S_E and E satisfy (*,**) to show that $R_{SE} \cup S_E$ is Church-Rosser under our assumptions that they satisfy the local properties, which indeed implies the desired property for both variants.

We now need to characterize the joinability properties of the critical patterns (i, ii, iii, iv).

For rewrite peaks, we need to check for joinability *complete sets of critical pairs of R modulo $S \cup E$*. This assumes that such complete sets exist. Note that it is possible to filter out the pairs which overlap is simplifiable by S_E.

For rewrite cliffs, we generate a normalized *E-extension* $g[l]_p\!\downarrow \to g[r]_p\!\downarrow$ for each rule $l \to r$ with respect to the equation $g = d \in E$ at $p \in \mathcal{FP}os(g) \setminus \{\Lambda\}$ provided l and $g|_p$ unify modulo $E \cup S$ (and symmetrically with d).

For simplification cliffs, we generate a normalized *oriented S-extension* $g[l]_p\!\downarrow \to g[r]_p\!\downarrow$ for each rule $l \to r$ with respect to the rule $g \to d \in S$ at $p \in \mathcal{FP}os(g) \setminus \{\Lambda\}$ provided l and $g|_p$ unify modulo $E \cup S$.

We are left with simplification peaks, which require a new kind of extension:

▶ **Definition 3.6.** Given rules $l \to r \in R$ and $g \to d \in S$, and a position $q \in \mathcal{FP}os(l)$ such that $l|_q$ and g are E-unifiable with a complete set of unifiers Σ, then the rules in $\{(l[d]_q)\sigma\!\downarrow \to (r\sigma)\!\downarrow \mid \sigma \in \Sigma$ are called *simplification pairs* of $g \to d$ onto $l \to r$ at position q.

▶ **Lemma 3.7.** *Assume (R, S, E) is closed under simplification pairs. Then critical simplification peaks are shallow-joinable.*

Proof. Note that $s \longrightarrow_{R_{S_E}} t$ for any simplification pair $s \to t$, hence R can be closed by these extensions without compromising soundness nor termination.

By monotonicity of rewriting, we can assume wlog that $p = \Lambda$. Let $u \xrightarrow{\Lambda}_R v$ with $l \to r \in R$ and $u \xrightarrow{q\in\mathcal{FP}os(l)\setminus\{\Lambda\}}_{S_E} w$ with $g \to d \in S$, and $w = l\tau[d]_q\tau$ (assuming $\mathcal{V}ar(l) \cap \mathcal{V}ar(g) = \emptyset$). Then, $u|_q = (l|_q)\tau =_E g\tau$. By closure assumption, some rule $l[g]\sigma{\downarrow} \to r\sigma{\downarrow}$ belongs to R for some σ such that $\tau =_E \sigma\theta$. Therefore, we get $w = (l[d]_q)\tau \xleftrightarrow[E]{(\geq_{\mathcal{P}}\ q)^*} (l[d]_q)\sigma\theta \xrightarrow{*}_{S_E} (l[d]_q)\sigma{\downarrow}\,\theta$ $\longrightarrow_R r\sigma{\downarrow}\,\theta\ {}_{S_E}\xleftarrow{*} r\tau = v$ and we are done. ◀

Note that we need to generate an extension for each substitution in $CSU(l|_q, g)$ rather than the single extension $l[d]_q \to r$ as for the other cases, since the latter would not yield shallow-joinability. On the other hand, we could require that simplification pairs satisfy shallow joinability, and indeed adding them as rules ensures that property.

We are now ready for the main result of this section:

▶ **Theorem 3.8.** *Let (R, S, E) be a NARS s.t.*
(i) *SE-critical pairs of R are joinable,*
(ii) *R is closed under normalized E-extensions,*
(iii) *R is closed under normalized simplification pairs, and*
(iv) *R is closed under normalized oriented S-extensions.*
Then normal rewriting is Church-Rosser.

Normal rewriting has many advantages: first, it allows rewriting with R modulo SE, despite the fact that congruence classes modulo SE may be infinite; second, compared to rewriting modulo SE, it allows to narrow down the sets of critical pairs and extensions; third, normal rewriting has a stronger rewriting power than normalized rewriting, and less critical pairs need be computed.

3.5 Example

We consider the example of the introduction: $R = \{x + x^{-1} \to 0\}$, $S = \{x + 0 \to x\}$ and $E = AC$. First, the termination assumption is satisfied. This is classically shown by a polynomial interpretation. Define $[\![x + y]\!] = [\![x]\!] + [\![y]\!]; [\![x^{-1}]\!] = 1 + [\![x]\!]$ and $[\![0]\!] = 0$. We then verify that equations in $S \cup E$ are invariant under the interpretation, while the rule in S decreases strictly. A lexicographic argument yields termination of $R_{SE} \cup S_E$.

We know that S is Church-Rosser modulo AC, which we did as an application of Theorem 3.5.

Finally, we need to show that normal rewriting with R is Church-Rosser. $x + x^{-1}$ unifies with the strict subterm $y + z$ of A, hence we need to add the extension $x + x^{-1} + y \to y$ (simplifying the righthand side $(0 + y)$ into y) to R. To ensure local confluence with S_E, we need to add $0^{-1} \to 0$ to R which has become $R = \{x + x^{-1} \to 0, (x + x^{-1}) + y \to y, 0^{-1} \to 0\}$. We can then verify that the resulting system satisfies our result, hence is Church-Rosser.

4 Higher-order rewriting systems

Our interest here is in higher-order rewriting as introduced by Nipkow [19, 17]. Nipkow and Mayr assume that rules in R are simply typed, of basic type, in η-long β-normal form and that their lefthand sides are patterns. Higher-order rules are fired via higher-order pattern matching. Other, related approaches to higher-order rewriting are considered and compared in [22].

We now consider a simply typed lambda calculus $\lambda^{\rightarrow}$ generated by sets of function symbols $\mathcal{F}_n$ with arity $n \geq 0$ and a set of variables $\mathcal{X}$, the type structure being itself generated by a set $\mathcal{S}$ of type constants. We use a, b for types, and s, t for (raw) terms:

$$\begin{array}{rcl} a, b & := & \mathcal{S} \mid a \rightarrow b \\ s, t & := & x \mid f(s_1, \ldots, s_n) \mid \lambda x : a \,.\, s \mid (s\, t) \end{array}$$

Typing judgements are of the form $\Gamma \vdash s : a$, where Γ is a set of declarations of the form $\{x_1 : a_1, \ldots, x_n : a_n\}$ such that $x_i \neq x_j$ if $i \neq j$. A term s is *typable* of type a in the environment Γ if the judgement $\Gamma \vdash s : a$ can be proved from the following rules:

var: $$\frac{}{\Gamma \cup x : a \vdash x : a}$$

fun: $$\frac{\Gamma \vdash s_1 : a_1, \ldots, \Gamma \vdash s_n : a_n}{\Gamma \vdash f(s_1, \ldots, s_n) : a}$$

`if` $f : a_1 \rightarrow \ldots \rightarrow a_n \rightarrow a \in \mathcal{F}_n$

abs: $$\frac{\Gamma \cup x : a \vdash s : b}{\Gamma \vdash (\lambda x : a \,.\, s) : a \rightarrow b}$$

app: $$\frac{\Gamma \vdash s : a \rightarrow b \quad \Gamma \vdash t : a}{\Gamma \vdash (s\, t) : b}$$

As usual, substitutions are *capture*-avoiding "morphism" written $\{x_1 \mapsto s_1, \ldots, x_n \mapsto s_n\}$ when finitely variables are involved, such that x_i and s_i have the same type in the environment Γ.

$\lambda^{\rightarrow}$ comes equipped with three equations:

alpha: $\lambda x : a \,.\, s = \lambda y : a \,.\, s\{x \mapsto y\}$ if $y \notin \mathcal{V}ar(s)$

beta: $((\lambda x : a \,.\, s)\, t) = s\{x \mapsto t\}$

eta: $\lambda x : a \,.\, (s\, x) = s$ if $x \notin \mathcal{V}ar(s)$

beta is oriented as a rule from left to right, while eta can be oriented as a reduction from left to right or as an expansion from right to left. The set S of simplifiers is made of beta and a choice of orientation for eta.

Let us now recall that the presence of binders forces to rewrite modulo α-conversion, even when rewriting with the beta rule alone. The simplest example is due to Barendregt and Klop:

$(\lambda x.(x\ x)\ \lambda s.\lambda z.(s\ z)) \longrightarrow (\lambda s.\lambda z.(s\ z)\ \lambda s.\lambda z.(s\ z)) \longrightarrow \lambda z.(\lambda s.\lambda z.(s\ z)\ z) \longrightarrow \lambda z.\lambda z.(z\ z)$.

which last step has resulted in the variable z being captured by λz. We should instead rename the *inside* binder $\lambda z.$, showing once more that rewriting modulo (here, α-conversion) surfaces everywhere:

$\lambda z.(\lambda s.\lambda z.(s\ z)\ z) \xleftrightarrow[\alpha]{(\geq 1)^*} \lambda z.(\lambda s'.\lambda z'.(s'\ z')\ z) \xrightarrow[\beta]{1} \lambda z.\lambda z'.(z\ z')$.

We now move to our higher-order rewrite rule format, which definition is the following:

▶ **Definition 4.1.** A *rewrite rule* is a tuple (Γ, l, r, σ) s.t.

(i) l and r are in S_E-normal form,

(ii) $\Gamma \vdash l : \sigma$ and $\Gamma \vdash r : \sigma$,

(iii) $l = f(l_1, \ldots, l_n)$ for some $f \in \mathcal{F}_n$, and is a pattern [18],

(iv) $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$.

We write $\Gamma \vdash l \rightarrow r : \sigma$ or simply $l \rightarrow r$ if no ambiguity.

It is actually not necessary to assume that lefthand sides of rules are headed by a function symbol, but, besides being a natural assumption, it simplifies the critical pairs analysis. We shall however explain what are the additional computations needed when this assumption is not met.

On the other hand, the pattern assumption cannot be dispensed with as pointed out to us by Vincent van Oostrom:
Given type constants o, i, let $\mathcal{F} = \{f : (o \to o) \to o, a : o, h : o \to i, g : i \to o\}$, $\mathcal{X} = \{o \to o, X : i\}$, and $R = \{f(\lambda x : o \,.\, F(F(x)) \to a, h(g(X)) \to X\}$.
Then $a \longleftarrow f(\lambda x.g(h(g(h(x))))) \longrightarrow f(\lambda x \,.\, g(h(x)))$ with both terms in normal form despite the fact that there are no critical pairs in the usual sense since $f(\lambda x \,.\, F(F(x)))$ has no subterm of type i. The role of the pattern restriction is indeed to rule out these non-intuitive higher-order phenomena.

The assumption that rules are in S_E-normal form is important as well to ensure the absence of simplification peaks in most cases.

We will use variations of a single example, differentiation, writing $u(v)$ instead of $(u\,v)$:

R	:	$*$	% type of reals
$\times^2$	:	$\mathsf{R} \to \mathsf{R} \to \mathsf{R}$	% arity 2
diff^1	:	$(\mathsf{R} \to \mathsf{R}) \to (\mathsf{R} \to \mathsf{R})$	% arity 1
$\mathsf{sin}^1, \mathsf{cos}^1$	:	$\mathsf{R} \to \mathsf{R}$	% arity 1
$\times^2_{\to}$	:	$(\mathsf{R} \to \mathsf{R}) \to (\mathsf{R} \to \mathsf{R}) \to (\mathsf{R} \to \mathsf{R})$	
F	:	$\mathsf{R} \to \mathsf{R}$	

$$\mathsf{diff}(\lambda x.\, \mathsf{sin}(F(x))) \to \lambda x.\, \mathsf{cos}(F(x)) \times_{\to} \mathsf{diff}(F)$$

The idea here is to embed composition into the definition, the usual rule for differentiating a sinus being recovered thanks to higher-order matching by instantiating F by the identity. Note that patterns occur naturally in examples.

Our termination assumption can be verified easily here by using the Normal Higher-Order Recursive path Ordering introduced in [13]. All the coming variants can be dealt with as well.

Although $\beta\eta$-congruence classes are infinite, termination of higher-order rules like the above one is usually easy to show because the top function symbols in the rules are different from those of the λ-calculus. Normal rewriting therefore appears to be a very good fit with higher-order computations.

4.1 Higher-order rewriting at simple types

4.1.1 η as an expansion [19, 17]

E is α-conversion and S is made of β-reduction and η-expansion saturating arrow types:

$$v[u]_p \longrightarrow_\eta v[\lambda x : a \,.\, (u\,x)]_p$$
$$\text{if} \begin{cases} u : a \to b \\ x \notin \mathcal{V}ar(u) \\ \text{if } p = q \cdot 1, \text{ then } v|_q \text{ is not an application} \end{cases}$$

To comply with the so-called Nipkow's format for which rules must operate at base types, the example becomes:

$$\mathsf{diff}(\lambda x.\, \mathsf{sin}(F(x)))(y) \to \mathsf{cos}(F(y)) \times \mathsf{diff}(\lambda x. F(x))(y)$$

There are of course no extensions associated with α-conversion.

Since η is used as an expansion, and has therefore a variable as its lefthand side, there are no oriented extensions associated with the η-rule, and no simplification extensions either, since rules in R are in normal form for the simplification rules.

There are no oriented extension for the β-rule since the rules in R being of base type, their lefthand side cannot unify with an abstraction, which is the sole strict non-variable subterm of the lefthand side of the beta rule. Note that the argument still holds for lefthand sides which are not headed by a function symbol. There are no simplification extensions either: since rules are in η-long β-normal form, no subterm of a rule can unify with the lefthand side of β except for subterms of the form $(X\,x)$, where X is a higher-order variable and x a variable bound above. Instantiating X by $\lambda y.u$ for some term u yields a term which is a higher-order instance of l, hence rewrites to the corresponding instance of the righthand side. It appears therefore that rules in R are their own simplification pairs. Therefore,

▶ **Theorem 4.2** ([17]). *Higher-order rewriting with R is Church-Rosser provided*
(i) $\longrightarrow_{R_{\beta\eta}} \cup \longrightarrow_{\beta\eta^{-1}}$ *is α-terminating;*
(ii) *irreducible higher-order critical pairs are joinable.*

where higher-order critical pairs are defined a usual by solving equations of the form $l|_p =_{\beta\eta\alpha} g$ for some rules $l \to r$ and $g \to d$. Note the strong analogy with Theorem 3.2. This is due to the choice of orienting η as an expansion, and to rule out user's rules at higher-type. Note also that the presence of α in our termination assumption, which is usually omitted (that is, left implicit).

This version of Nipkow's result requires to prove that the relation $(\longrightarrow_{R_{\beta\eta}} \cup \longrightarrow_{\beta\eta^{-1}}$ is α-terminating, instead of $\longrightarrow_{R_{\beta\eta}}$ and $\longrightarrow_{\beta\eta^{-1}}$ being separately α-terminating as in Nipkow's original result. On the other hand, we conclude for the stronger Church-Rosser property instead of confluence as does Nipkow. Note also the little improvement, compared with Nipkow's result, obtained by eliminating the reducible higher-order critical pairs from the joinability test.

4.1.2 η as a reduction:

S is now made of β and η reductions. We will nevertheless recover the advantages of η-expansions by having arities for the variables as in Klop's framework: they can be η-expanded up to the saturation of their arity, as in $\lambda xy.X(x,y)$ for X of arity two and x, y of arity zero. This η-expanded term is indeed η-reduced, since $X(x)$ and X are not terms in this setting.

The only difference with the previous case is therefore the orientation of η. Since simplification pairs require the unification (modulo α) of the lefthand side of the η-rule with a subterm of a lefthand side of R, the only potential case is that of a subterm of a lefthand side of rule in R being of the form $\lambda x.X(x)$ where X is a free higher-order variable. Rewriting this subterm with η is not possible, though, since it would violate the arity of X. We are therefore left with oriented extensions, for which a lefthand side of rule would unify the only non-variable strict-subterm of the lefthand side of η, which is impossible with our rule format (which could actually be relaxed).

▶ **Theorem 4.3.** *Higher-order rewriting with R is Church-Rosser provided*
(i) $\longrightarrow_{R_{\beta\eta}} \cup \longrightarrow_{\beta\eta}$ *is α-terminating;*
(ii) *irreducible higher-order critical pairs are joinable.*

This shows that the choice of orienting eta as a reduction or an expansion has no impact on confluence when rewriting is only possible at simple types.

4.2 Higher-order rewriting at higher types

To understand the importance of the type assumption in Nipkow's format, let us consider his motivating example $R = \{\lambda x.a \to \lambda x.b\}$, where a and b are constants of a given base type. a and b are convertible terms in η-long β-normal form since $a \xleftrightarrow[\beta]{\Lambda} (\lambda x.a\ u) \xleftrightarrow[R]{1} (\lambda x.b\ u) \xleftrightarrow[\beta]{\Lambda} b$, but not joinable.

This has motivated Nipkow's restriction that the lefthand side of a higher-order rule is of base type, and therefore, is not an abstraction. Of course, it would be easy to change the rule into $a \to b$ (making it satisfy our definition of rule), therefore avoiding the problem, but this cannot be done in general. Consider for instance the rewrite rule $\lambda x.\, f(x, Z(x)) \to \lambda x.\, g(x, Z(x))$. Removing the abstraction yields $f(X, Z(X)) \to g(X, Z(X))$, a rule which left-hand side is no pattern.

Nipkow's example shows a case of critical simplification peak: adding the normalized β-extension of the original rule $\lambda x.a \to \lambda x.b$, that is $a \to b$ solves the problem. In general, the normalized β-extension of a rule $\lambda x.l(x) \to r$ is the rule $l(x) \to @(r, x)\downarrow$. Since one abstraction is pulled out, a rule can have only finitely many such extensions which are easily computable.

Our format rules out the need for these extensions, since lefthand sides cannot be abstractions. The previous argument that simplification pairs are not needed remains valid. Therefore,

▶ **Theorem 4.4.** *Higher-order rewriting with R is Church-Rosser provided*
(i) $\longrightarrow_{R_{\beta\eta}} \cup \longrightarrow_{\beta\eta^{-1}}$ *is α-terminating,*
(ii) *irreducible higher-order critical pairs are joinable.*

We can therefore reformulate our example as follows:

$$\begin{array}{lcl} \mathsf{diff}^1 & : & (\mathsf{R} \to \mathsf{R}) \to (\mathsf{R} \to \mathsf{R}) \\ \mathsf{sin}^0, \mathsf{cos}^0 & : & \to (\mathsf{R} \to \mathsf{R}) \\ F^0 & : & \to (\mathsf{R} \to \mathsf{R}) \\ \times^1_{\to} & : & (\mathsf{R} \to \mathsf{R}) \to (\mathsf{R} \to \mathsf{R}) \\ \multicolumn{3}{l}{\mathsf{diff}(\mathsf{sin} \circ F) \to \mathsf{diff}(\mathsf{sin}) \times^1_{\to} \mathsf{diff}(F)} \end{array}$$

Orienting eta as a reduction would yield the same result again for our rule format.

4.3 Adding algebraic equations in E

Our main abstract result allows us to also consider equations like AC in E which would then contain both AC and α, see [6] for examples. Of course the presence of AC requires checking new pairs for the corresponding local properties. Higher-order unification modulo AC of higher-order patterns yields finite complete sets of unifiers [6], hence the calculations which require higher-order unification yield decidable tests when lefthand sides of rules are patterns.

5 Conclusion

We have given a framework for normal rewriting terms that covers a wide variety of rewriting applications, whether first or higher-order. These results are very economic thanks to an abstract framework which incorporates a lightweight axiomatization of positions. Besides, they solve a long-standing open problem regarding how to check the Church-Rosser property of higher-order rewriting at any type, whether basic or functional.

We believe that the application of our main abstract result to higher-order rewriting can be pushed further, by allowing for polymorphic or dependent types.

A referee strongly suggested many potential extensions of the framework, to a fully hierarchical setting, to non-terminating normal rewriting (using decreasing diagrams), to graph rewriting, and more. Although we are not interested ourselves in those extensions at this point, we would of course welcome efforts in these directions. Such extensions would provide the appropriate theoretical basis for several recent applications, the most surprising to us being [1].

Acknowledgements: to Delia Kesner, Femke van Raamsdonk and Albert Rubio for discussing actively this topic in the past.

References

1 O. Al-Hassani, Q.-A. Mahesar, C. Sacerdoti Coen, and V. Sorge. A term rewriting system for Kuratowski's closure-complement problem. In Tiwari A., editor, *RTA*, volume 15 of *LIPIcs*, pages 38–52. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2012.

2 L. Bachmair. *Canonical Equational Proofs*. Birkhäuser, Boston, 1991.

3 L. Bachmair, N. Dershowitz, and J. Hsiang. Orderings for equational proofs. In *Proc. 1st LICS*, 1986.

4 T. Baird, Peterson G. E., and Wilkerson R. Complete sets of reductions modulo Associativity, Commutativity and Identity. In *Proc. 3rd RTA, LNCS 355*, pages 29–44. Springer, 1989.

5 H. Barendregt. *Handbook of Theoretical Computer Science*, volume B, chapter Functional Programming and Lambda Calculus. North-Holland, 1990.

6 A. Boudet and E. Contejean. AC-unification of higher-order patterns. In *Int. Conf. on Constraint Programming, 1997*, pages 267–281, 1997.

7 N. Dershowitz and J.-P. Jouannaud. *Handbook of Theoretical Computer Science*, volume B, chapter Rewrite Systems. North Holland, 1990.

8 G. Gonthier, J.-J. Lévy, and P.-A. Mellies. An abstract standardisation theorem. In *LICS*, pages 72–81. IEEE Computer Society, 1992.

9 G. Huet. Confluent reductions: abstract properties and applications to term rewriting systems. *Journal of the ACM*, 27(4):797–821, October 1980.

10 J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computing*, 15(4):1155–1194, 1986.

11 J.-P. Jouannaud and C. Marché. Termination and completion modulo associativity, commutativity and identity. *Theoretical Computer Science*, 104:29–51, 1992.

12 J.-P. Jouannaud and F. van Raamsdonk. Confluence properties of terminating higher-order rewrite relations. In *Mathematical Theories of Abstraction, substitution and naming in Computer Science*, ICMS, Edinburgh, may 2007.

13 J.-P. Jouannaud and A. Rubio. Higher-order orderings for normal rewriting. In F. Pfenning, editor, *RTA*, volume 4098 of *LNCS*, pages 387–399. Springer, 2006.

14 D. E. Knuth and P. B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*. Elsevier, 1970.

15 D. S. Lankford and Ballantyne A. M. Decision procedures for simple equational theories with commutative-associative axioms: Complete sets of commutative-associative reductions. Memo ATP-39, University of Texas, Austin, August 1977.

16 C. Marché. Normalised rewriting and normalised completion. In *Proc. 9th LICS*, 1994.

17 R. Mayr and T. Nipkow. Higher-order rewrite systems and their confluence. *Theoretical Computer Science*, 192:3–29, 1998.

18 D. Miller. A logic programming language with lambda-abstraction, function variables, and simple unification. *Journal of Logic and Computation*, 1(4):497–536, 1991.

19 T. Nipkow. Higher-order critical pairs. In *Proceedings of the 6th annual IEEE Symposium on Logic in Computer Science (LICS '91)*, pages 342–349, Amsterdam, The Netherlands, July 1991.

20 V. van Oostrom. Confluence by decreasing diagrams converted. In Voronkov A., editor, *RTA*, volume 5117 of *Lecture Notes in Computer Science*, pages 306–320. Springer, 2008.

21 G. E. Peterson and M. E. Stickel. Complete sets of reductions for some equational theories. *Journal of the ACM*, 28(2):233–264, April 1981.

22 F. van Raamsdonk. Higher-order rewriting. In P. Narendran and M. Rusinowitch, editors, *Proc. RTA*, LNCS 1631, pages 220–239. Springer, 1999.

23 Terese. Term rewriting systems. In *Cambridge Tracts in Theoretical Computer Science, M. Bezem, J. W. Klop, and R. de Vrijer editors*. Cambridge University Press, 2003.