Confluence of conditional rewrite systems — Source link

Nachum Dershowitz, Mitsuhiro Okada, G. Sivakumar

Institutions: University of Illinois at Urbana–Champaign, Concordia University

Published on: 01 Aug 1988

Topics: Confluence, Rewriting and Algebraic data type

Confluence of Conditional Rewrite Systems*

Nachum Dershowitz[1], Mitsuhiro Okada[2], and G. Sivakumar[1]

[1]Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801, U.S.A.

[2]Department of Computer Science
Concordia University
Montreal, Quebec H3G 1M8, Canada

ABSTRACT

Conditional rewriting has been studied both from the point of view of algebraic data type specifications and as a computational paradigm combining logic and functional programming. An important issue, in either case, is determining whether a rewrite system has the *Church-Rosser*, or *confluence*, property. In this paper, we settle negatively the question whether ''joinability of critical pairs'' is, in general, sufficient for confluence of terminating conditional systems. We review known sufficient conditions for confluence, and also prove two new positive results for systems having critical pairs and arbitrarily big terms in conditions.

1. Introduction

Conditional term rewriting systems arise naturally in the algebraic specification of data types; they have been studied largely from this perspective [Zhang-Rémy-85, Bergstra-Klop-86, Kaplan-87]. When a system is *noetherian* and *confluent*, it defines a normal-form algebra which is initial. With various restrictions on the form and content of conditional rules, e.g. only one occurrence of any variable on the left-hand side, some useful conditions for confluence have been obtained. In this paper, we extend those results.

Recently, conditional rewriting systems have been shown to provide a natural computational paradigm combining logic and functional programming [Fribourg-85, Goguen-Meseguer-86, Dershowitz-Plaisted-87]. A program is a set of conditional rules and a computation attempts to find a substitution that makes two terms provably equal in the underlying theory. The operational mechanisms are *rewriting* and *narrowing*: rewriting provides for the functional part and narrowing for the logic-programming ability to construct solutions. For the completeness of interpreters based on those mechanisms, some form of confluence is needed. In this context, it is especially useful to weaken any restrictions on the form of rules as much as possible, and still have syntactic tests for confluence. Unfortunately, as we show here, there is a limit to the extent to which such restrictions can be lifted.

* This research was supported in part by the National Science Foundation under Grant DCR 85-13417.

The next section presents the basic notions. In Section 3 we give an interesting counter-example to show that critical pair joinability is, in general, insufficient for confluence. Sections 4 and 5 contain our positive results on confluence. We conclude with suggestions of extensions and applications.

2. Conditional Rewriting

A *positive conditional equation* is a (first-order) formula of the form

$$p_1 = q_1 \wedge \cdots \wedge p_n = q_n \supset l = r,$$

where $l, r, p_1, q_1, \ldots, p_n$, are q_n are (first-order) terms built from some set F of function symbols and set X of variables. The formula is (implicitly) universally quantified. The subformulae $p_1 = q_1, \ldots, p_n = q_n$ $(n \geq 0)$ are called *conditions*; an equation is *unconditional* if $n = 0$. We will abbreviate conditions by writing, instead, $\overline{p} = \overline{q} : l = r$.

A *conditional rewrite rule* is an oriented conditional equation, written

$$\overline{p} \downarrow \overline{q} : l \rightarrow r,$$

or just $p \downarrow q : l \rightarrow r$, where the symbol $\downarrow$ signifies equality and has an operational meaning described below. A *conditional rewrite system* is a set of such rules. For example, the following is a system for computing the relation $\leq$ on natural numbers:

$$\begin{array}{rrcl} & 0 \leq 0 & \rightarrow & tt \\ & s(x) \leq 0 & \rightarrow & ff \\ & s(x) \leq s(y) & \rightarrow & x \leq y \\ x \leq y \downarrow tt : & x \leq s(y) & \rightarrow & tt \end{array}$$

Conditional systems are used to compute by replacing an instance of a left-hand side l by the corresponding instance of the right-hand side r, provided the corresponding instance of the condition $p \downarrow q$ holds.

For a given system, we define the *rewrite* ($\rightarrow$) and *join* ($\downarrow$) relations on terms, as follows: Let $p \downarrow q : l \rightarrow r$ be a rule, s be a term, π be a position of a subterm in s, and σ be a substitution (a mapping from variables to terms, extended to a morphism from terms to terms). Then we say that the term $s[l\sigma]_\pi$, that is, the term s with an instance $l\sigma$ of the left-hand side l at position π, *rewrites* to the term $s[r\sigma]_\pi$ (s with $r\sigma$ in place of $l\sigma$) if $p\sigma$ and $q\sigma$ each rewrite in zero or more steps to the identical term; in that case, we say that σ is a *feasible* substitution for the rule. We write $s \rightarrow t$, if s rewrites to t in one step; $s \xrightarrow{*} t$, if s rewrites to t in zero or more steps, i.e. if t is *derivable* from s; $s \downarrow t$, if $s \xrightarrow{*} w$ and $t \xrightarrow{*} w$ for some term w; and $s \xrightarrow{!} t$, if $s \xrightarrow{*} t$, but no rewrite applies to t, i.e. the *normal form* t is derivable from s. For the above example, we have $0 \leq s(0) \xrightarrow{!} tt$ using the last rule, since the condition $0 \leq 0 \downarrow tt$ is achieved by the first rule. See [Dershowitz-Okada-Sivakumar-87] for an analysis of the impact of different operational semantics for conditions.

Two important restrictions on conditional rules are the following:

Definition 1. A conditional rewrite system is *left-linear* if a variable occurs at most once in a left-hand side l of each rule $\overline{p} \downarrow \overline{q} : l \rightarrow r$.

Definition 2. A conditional rewrite system is *normal* if every component of the right part $\overline{q}$ of the condition of each rule $\overline{p} \downarrow \overline{q} : l \rightarrow r$ is a ground (variable-free) normal form.

Note that for normal systems the condition $\overline{p} \downarrow \overline{q}$ is equivalent to $\overline{p} \xrightarrow{!} \overline{q}$.

Any conditional system can be transformed into a normal, but not left-linear, one, by introducing new operators, *eq* and *true*, writing conditions $\overline{p} \downarrow \overline{q}$ as $eq(\overline{p},\overline{q}) \downarrow true$, and adding the non-left-linear unconditional rule $eq(x,x) \rightarrow true$. A system that is both left-linear and normal is called *Type* III_n in [Bergstra-Klop-86].

A rewrite relation $\rightarrow$ is said to be *noetherian* if there is no infinite chain of terms $t_1, t_2, \cdots, t_k, \cdots$ such that $t_i \rightarrow t_{i+1}$ for all $i \geq 1$. A rewrite relation is *confluent* if whenever two terms, s and t, are derivable from a term u, then a term v is derivable from both s and t. In symbols: $\overset{*}{\leftarrow} \circ \overset{*}{\rightarrow} \subseteq \overset{*}{\rightarrow} \circ \overset{*}{\leftarrow}$, where $\overset{*}{\leftarrow}$ is the inverse of the derivability relation $\overset{*}{\rightarrow}$. It is *locally confluent*, if $s \overset{*}{\rightarrow} v$ and $t \overset{*}{\rightarrow} v$ for some v whenever $u \rightarrow s$ and $u \rightarrow t$ (in one step).

2.1. Disjoint peaks

If a term $u[s,t]$ contains subterms, s and t, neither of which is a subterm of the other, and each rewrites to another term, say $s \rightarrow s'$ and $t \rightarrow t'$, then the resultant terms, $u[s',t]$ and $u[s,t']$, rewrite in one step to the same term, $u[s',t']$. The situation $u[s',t] \leftarrow u[s,t] \rightarrow u[s,t']$ is called a *disjoint peak*. See Fig. 2.1.

2.2. Critical peaks

Throughout this paper, variables in different rules (or in different instances of the same rule) are considered to be distinct, renaming, as necessary.

Definition 3. If the left-hand side g of a rule $p' \downarrow q' : g \rightarrow d$ unifies, via most general substitution σ, with a non-variable subterm s at position π in a left-hand side l of a rule $p \downarrow q : l \rightarrow r$, then the conditional equation

$$(p\sigma, p'\sigma) = (q\sigma, q'\sigma): \ \ l\sigma[d\sigma]_\pi = r\sigma$$

is called a *critical pair* of the two rules, where $l\sigma[d\sigma]_\pi$ is obtained by replacing s in l by d and applying σ.

The situation

$$l\sigma[d\sigma]_\pi \leftarrow l\sigma \rightarrow r\sigma$$

is called a *critical overlap*, and, for any context u and substitution τ less general than σ,

$$u[l\tau[d\tau]_\pi] \leftarrow u[l\tau] \rightarrow u[r\tau]$$

is called a *critical peak*. See Fig. 2.2.

Definition 4. A critical pair $c = d : s = t$ is *feasible*, and the corresponding overlap is *feasible*, if there is a substitution σ for which $c\sigma \downarrow d\sigma$. A *trivial* critical pair is one for which s is identical to t. A system is *non-overlapping* if it has no feasible, non-trivial critical pairs.

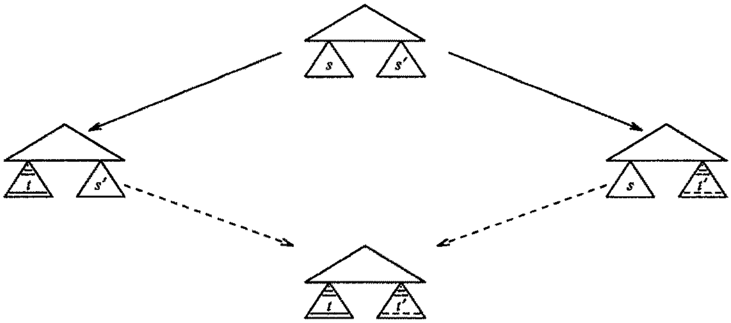

Figure 2.1. Disjoint peak.

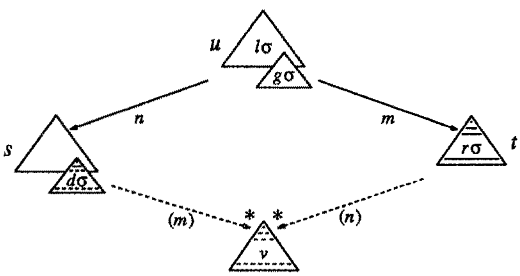

Figure 2.2. Joinable (Shallow-joinable) critical overlap.

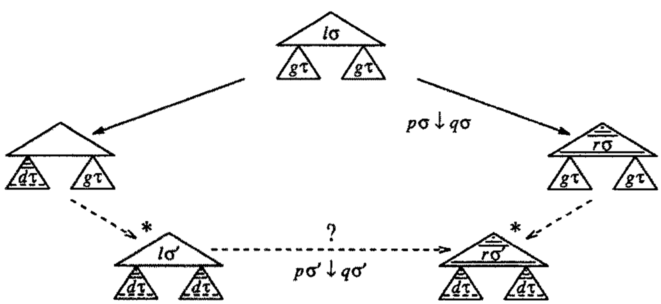

Figure 2.3. Variable overlap.

Definition 5. A critical pair $c = d : s = t$ is *joinable* if $s\sigma \downarrow t\sigma$ for any substitution σ such that $c\sigma \downarrow d\sigma$.

Infeasible critical pairs are vacuously joinable and trivial ones are trivially joinable.

The *depth* of a rewrite is the depth of recursive evaluations of conditions needed to determine that the matching substitution is feasible. Formally, the depth of an unconditional rewrite is 0; the depth of a rewrite using a conditional rule $p \downarrow q : l \rightarrow r$ and substitution σ is $depth(p\sigma{\downarrow}q\sigma) + 1$; the depth of a n-step derivation $s \xrightarrow{*} t$ is the maximum of the depths of each of the n steps; the depth of a "valley" $s \downarrow t$, joining at a term v, is the maximum of the depths of $s \xrightarrow{*} v$ and $t \xrightarrow{*} v$; and the depth of a zero-step derivation or valley is 0. We write $s \underset{k}{\rightarrow} t$ if $s \rightarrow t$ and the depth of the rewrite step is no more than k. Similarly $s \underset{k}{\xrightarrow{*}} t$ will mean that the maximum depth in that derivation is at most k. For example, $0 \le 0 \underset{0}{\rightarrow} tt$, $0 \le s(0) \underset{1}{\rightarrow} tt$, and $0 \le s^n(0) \underset{m}{\rightarrow} tt$, for all $m \ge n$.

Definition 6. A critical pair $c = d : s = t$, obtained from a critical overlap $s \leftarrow u \rightarrow t$ is *shallow-joinable*, if, for each substitution σ such that $c\sigma \downarrow d\sigma$, there exists a term v such that $s\sigma \underset{n}{\xrightarrow{*}} v$ and $t\sigma \underset{m}{\xrightarrow{*}} v$, where m is the depth of $u \rightarrow s$ and n is the depth of $u \rightarrow t$. A conditional rewrite system is *shallow-joinable* if each of its critical pairs is.

In other words, every critical pair of a shallow-joinable system joins with the corresponding depths less or equal to those of the critical overlap. (See Fig. 2.2.) In particular, critical pairs between unconditional rules must be unconditionally joinable.

2.3. Variable peaks

Let τ be a feasible substitution for a rule $p' \downarrow q' : g \rightarrow d$ and let σ be a feasible substitution for a rule $p \downarrow q : l \rightarrow r$ under which some variable x in l is mapped to a term $c[g\tau]$, containing an instance of g. Then, the term $l\sigma$ can be rewritten in two different ways, to $r\sigma$ and $l\sigma[g\tau]$, as depicted in Fig. 2.3. This is what we will call a *variable peak.*

If l is non-linear in x, then each of the remaining occurrences of $c[g\tau]$ in $l\sigma$ may be rewritten until a term $l\sigma'$ is obtained, where σ' is same as σ except that x is mapped to $c[d\tau]$. Similarly, if r is non-linear in x then we need additional rewrites to get $r\sigma \xrightarrow{*} r\sigma'$. When dealing with unconditional rewriting systems, variable peaks are always joinable, since $l\sigma' \rightarrow r\sigma'$. But for conditional systems, σ' must be feasible, i.e. $p\sigma' \downarrow q\sigma'$ must hold. This is not always the case, even if critical pairs are joinable, as the counter-examples of the next section demonstrate.

3. Counter-examples

In this section, we present non-confluent systems that are counter-examples to attempts at extending theorems for unconditional systems to the analogous conditional case.

Unconditional systems are locally confluent, if all their critical pairs are joinable. On the other hand, Example A [Bergstra-Klop-86] below shows that non-normal, non-noetherian conditional systems need not be locally confluent, even if they are left-linear and non-overlapping. In that example, the term b has many normal forms, including a and $f(a)$, despite the lack of critical pairs.

As will be shown in Section 5, noetherian conditional systems with no critical pairs are locally confluent, and, as is well known, for noetherian systems, local confluence implies confluence. Unfortunately:

	b	$\rightarrow$	$f(b)$
$x \downarrow f(x)$:	$f(x)$	$\rightarrow$	a

Example A. Left-linear and non-overlapping, but not normal.

	c	$\rightarrow$	$k(f(a))$
	c	$\rightarrow$	$k(g(b))$
	a	$\rightarrow$	b
	$h(x)$	$\rightarrow$	$k(x)$
	$h(f(a))$	$\rightarrow$	c
$h(f(x)) \downarrow k(g(b))$:	$f(x)$	$\rightarrow$	$g(x)$

Example B. Left-linear and normal, but not shallow-joinable.

$b \downarrow b$:	a	$\rightarrow$	b
$b \downarrow b$:	$h(x)$	$\rightarrow$	$k(x)$
$b \downarrow b$:	$h(f(a))$	$\rightarrow$	$p(a)$
	$p(b)$	$\rightarrow$	$k(f(a))$
	$q(b)$	$\rightarrow$	$k(g(a))$
$x \downarrow a$:	$p(x)$	$\rightarrow$	$q(x)$
$h(f(x)) \downarrow k(g(b))$:	$f(x)$	$\rightarrow$	$g(x)$

Example C. Left-linear and shallow-joinable, but not normal.

	$eq(x,x)$	$\rightarrow$	$true$
$b \downarrow b$:	a	$\rightarrow$	b
$b \downarrow b$:	$h(x)$	$\rightarrow$	$k(x)$
$b \downarrow b$:	$h(f(a))$	$\rightarrow$	$p(a)$
	$p(b)$	$\rightarrow$	$k(f(a))$
	$q(b)$	$\rightarrow$	$k(g(a))$
$eq(x,a) \downarrow true$:	$p(x)$	$\rightarrow$	$q(x)$
$h(f(x)) \downarrow k(g(b))$:	$f(x)$	$\rightarrow$	$g(x)$

Example D. Normal and shallow-joinable, but not left-linear.

Proposition. *There exists a noetherian, non-locally-confluent, conditional rewrite system all of whose critical pairs are joinable.*

This is demonstrated by Example B, in which, though all four critical pairs are joinable, the term $f(a)$ has two normal forms, $f(b)$ and $g(b)$. Note that the unconditional critical pair obtained by rewriting c in two ways is joinable only using the conditional rule, i.e. it is not shallow-joinable; see Fig. 3.1. With slight modifications, one obtains counter-examples C and D, showing that no combination of two of the following three factors suffices for confluence: left-linear, normal, and shallow-joinable. From these examples, it is clear that we need relatively strong restrictions on rewrite systems to guarantee confluence. In the next section, we show that combining all three factors does, in fact, yield confluence for noetherian systems.

4. Confluence of normal systems

In this section, we consider conditions that ensure that a normal, left-linear system is confluent. Such systems arise naturally in pattern-directed functional languages, when the different cases are constructor-based and mutually exclusive.

Bergstra and Klop have shown the following for conditional systems that are not necessarily noetherian:

Theorem 1 [Bergstra-Klop-86]. *A left-linear, normal conditional rewrite system is confluent, if it is non-overlapping.*

(Though we have weakened their definition of non-overlapping to allow infeasible overlaps, the result still holds.) This is analagous to the standard result that left-linear unconditional systems with no critical pairs are confluent (see, e.g. [Huet-80]), and can be extended somewhat by severely limiting the way in which critical pairs join.

We give a similar result for overlapping systems in which critical pairs are shallow-joinable. For this, we must require that the system be noetherian.

Theorem 2. *A noetherian, left-linear, normal conditional rewrite system is confluent, if all its critical pairs are shallow joinable.*

From the counter-examples of the previous section, one can see that this is optimal.

This theorem is a corollary of the following:

Lemma 1. *Let R be a noetherian conditional rewrite system that is left-linear, normal and shallow-joinable. Then, if $u \underset{m}{\overset{*}{\rightarrow}} s$ and $u \underset{n}{\overset{*}{\rightarrow}} t$, there exists a term v such that $s \underset{n}{\overset{*}{\rightarrow}} v$ and $t \underset{m}{\overset{*}{\rightarrow}} v$.*

Proof. The proof is by induction on the pair $(m+n, u)$ with respect to the (lexicographic combination of the) natural ordering of natural numbers and the noetherian relation $\rightarrow$ on terms.

Let $u \underset{m}{\rightarrow} s' \underset{m}{\overset{*}{\rightarrow}} s$ and $u \underset{n}{\rightarrow} t' \underset{n}{\overset{*}{\rightarrow}} t$. That is, u is first rewritten at position π to s' using rule $p \downarrow N : l \rightarrow r$ with depth no greater than m, and at position π' to t' using $q \downarrow M : g \rightarrow d$ with maximum depth n (M and N are normal forms). We show that s' and t' are joinable with appropriate depths at some term w. As in the Diamond Lemma for unconditional systems, two inductions (at the peaks, s' and t') show that s and t are also joinable with suitable depths.

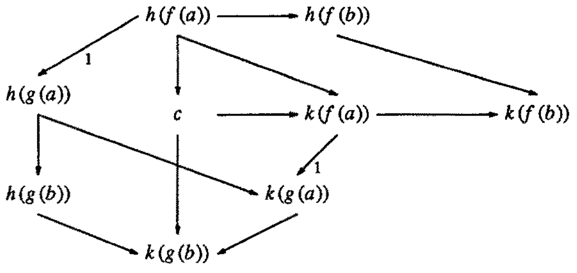

Figure 3.1. Critical pairs of Example B.

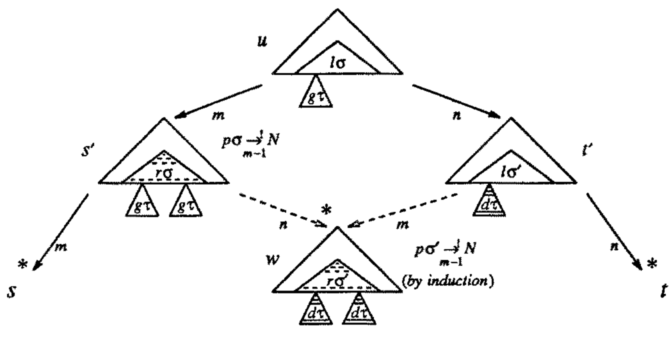

Figure 4.1. Normal, left-linear, variable overlap case (Lemma 1).

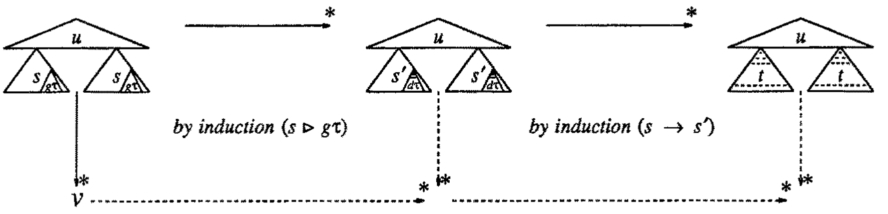

Figure 5.1. Proper subterm case (Lemma 2).

If the peak at u is *disjoint*, then s' and t' join at a term w (which is u after both rewrites). That is, $t' \underset{n}{\rightarrow} w$ (by rewriting at π') and $s' \underset{m}{\rightarrow} w$ (by rewriting at π).

If the peak is *critical*, then, by the shallow-joinable assumption, there is a term w such that $t' \overset{*}{\underset{n}{\rightarrow}} w$ and $s' \overset{*}{\underset{m}{\rightarrow}} w$.

This leaves only the *variable peak* case. Without loss of generality, let π be above π'. Thus, some variable x in l matches a subterm $c[g\tau]$ which rewrites to $c[d\tau]$. Let σ' be the same as the substitution σ used in rewriting $u \rightarrow s'$, except that x is mapped to $c[d\tau]$. As seen in Fig. 4.1, because R is left-linear, the subterm of t' at π is actually $l\sigma'$. Furthermore, $s' \overset{*}{\underset{n}{\rightarrow}} u[r\sigma']_{\pi} = w$, by rewriting all (zero or more) occurrences of $g\sigma$ in s' to $d\sigma$. It remains to show that $l\sigma' \underset{m}{\rightarrow} r\sigma'$ is feasible. Since the system is normal, we have $p\sigma' \overset{!}{\underset{m-1}{\rightarrow}} N$, as is required, by induction from the shallower peak $p\sigma' \overset{*}{\underset{n}{\leftarrow}} p\sigma \overset{!}{\underset{m-1}{\rightarrow}} N$. □

5. Confluence of joinable systems

In this section, we make no restrictions on the joinability of the critical pairs. We also do not insist on normal conditions or left-linearity. Under certain circumstances, we are able to prove that such systems are confluent as long as all their critical pairs are joinable. This is close in spirit to the result for unconditional systems.

> **Definition 7.** A system is *decreasing* if there exists a well-founded ordering $>$, containing the proper subterm ordering $\triangleright$, such that $s>t$ whenever $s \rightarrow t$ and, for each rule $p \downarrow q : l \rightarrow r$ and substitution σ, $l\sigma > p\sigma$ and $l\sigma > q\sigma$.

This is stronger than the noetherian requirement. Examples B-D are not decreasing, since the left-hand side $f(x)$ is a proper subterm of the term $h(f(x))$ in the condition.

> **Theorem 3** [Dershowitz-Okada-Sivakumar-87]. *A decreasing conditional rewrite system is confluent, if all its feasible critical pairs are joinable.*

This generalizes results in [Kaplan-87] and [Jouannaud-Waldmann-86], but the proof is essentially the same.

In our last result, we allow arbitrary terms in the condition, but insist that overlaps between left-hand sides do not occur at proper subterms of the overlapped left-hand side. In particular, non-deterministic pattern-directed languages, with no nested defined function symbols in the patterns, meet this requirement.

> **Definition 8.** A critical pair is an *overlay* if it is obtained from two two left-hand sides that unify at their roots.

In our original example, the critical pair $s(x) \le y = tt : x \le y = tt$ between the rules $s(x) \le s(y) \rightarrow x \le y$ and $x \le y \downarrow tt : x \le s(y) \rightarrow tt$ is an overlay.

> **Theorem 4.** *A noetherian conditional rewrite system is confluent, if all its critical pairs are joinable overlays.*

This may be useful for designing completion procedures, because of the absence of restrictions on conditions. It is worth noting that, in particular, any conditional system that is noetherian and non-overlapping is confluent, as is the case for unconditional systems.

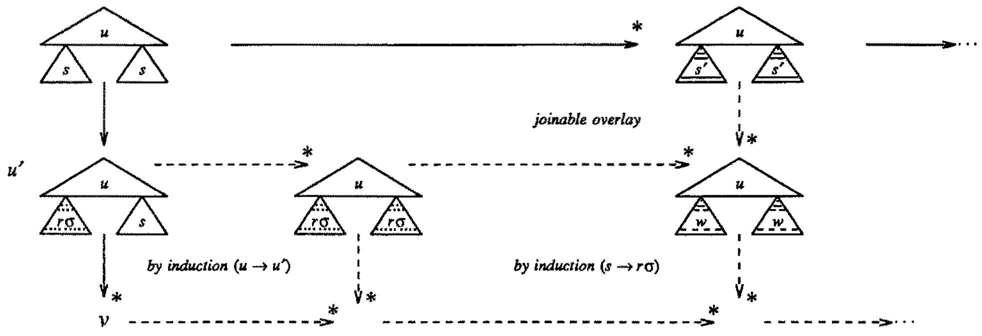

Figure 5.2. Overlay case (Lemma 2).

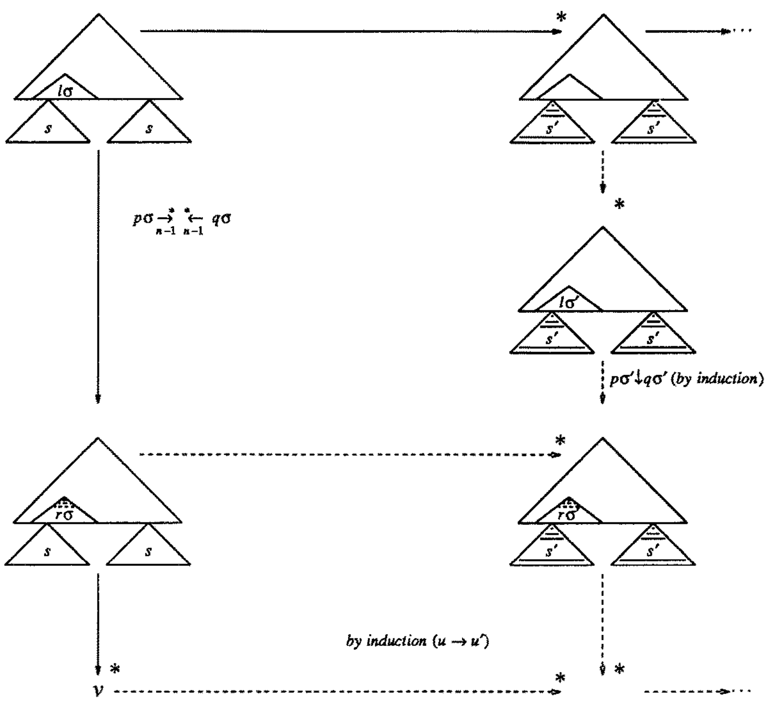

Figure 5.3. Variable overlap above (Lemma 2).

This theorem is a consequence of the following:

Lemma 2. *Let $u[t]_\Pi$, where Π is a set of positions, denote the term u with each subterm at a position in Π replaced by t. Let R be a noetherian conditional rewrite system in which all critical pairs are joinable overlays. If $u \xrightarrow{*} v$ and $s \xrightarrow{*} t$ for any terms u, v, s, and t, then $u[t]_\Pi \downarrow v$.*

Proof. We show that if $u[s]_\Pi \underset{n}{\xrightarrow{*}} v$ and $s \xrightarrow{*} t$, then $u[t]_\Pi \downarrow v$, by induction on the triple $(s\ ,\ n\ ,\ u[s])$, where the first component is compared using the union of the noetherian rewrite relation $\rightarrow$ and the proper subterm relation $\triangleright$, the second as a natural number, and the third by the rewrite relation.

If $u = v$ or $s = t$, then we are done. Suppose $s \rightarrow s' \xrightarrow{*} t$. If we can show that $u[s']_\Pi \downarrow v$, then by induction it will follow that $u[t]_\Pi \downarrow v$, since s' is less (vis-a-vis $\rightarrow$) than s.

If the first rewrite $s \rightarrow s'$ occurs at a proper subterm $g\tau$ of s, then by induction on the first component, we have $u[s']_\Pi \downarrow v$. See Fig. 5.1.

Otherwise, we may suppose that s is $g\sigma$ and s' is $d\sigma$, for some rule $p' \downarrow q' : g \rightarrow d$. Let $u[s] \underset{n}{\rightarrow} u' \underset{n}{\xrightarrow{*}} v$, with the first step via rule $p \downarrow q : l \rightarrow r$ at position π. If this is a disjoint peak (i.e. if π is neither above nor below any position in Π), then $u[s']_\Pi$ rewrites (at position π) to $u'[s']_\Pi$. Since u' is smaller than u, we have that $u'[s']_\Pi \downarrow v$. Thus, $u[s']_\Pi \rightarrow u'[s']_\Pi \downarrow v$.

If $u' \leftarrow u[s]_\Pi \xrightarrow{*} u[s']_\Pi$ is a critical peak, then it must be an overlay and $s = g\tau = l\sigma$. Critical pairs are joinable, so let $s' = d\tau \xrightarrow{*} w$ and $r\sigma \xrightarrow{*} w$, for some w. Then we have that $u' = u[r\sigma]_\pi \xrightarrow{*} u[r\sigma]_\Pi$. By induction on the last component, we have that $u[r\sigma]_\Pi \downarrow v$; by induction on the first, we get $u[w]_\Pi \downarrow v$. Thus, $u[s']_\Pi \xrightarrow{*} u[w]_\Pi \downarrow v$. This case is depicted in Fig. 5.2.

The remaining case is that of a *variable overlap*, either above or in some s. Let π be above s; that is, some variable x in l matches a term $c[s]_{\Pi'}$ containing any number of occurrences of s. Let σ' be the same as the substitution σ used to rewrite $u \rightarrow u'$, except that x is mapped instead to $c[s']_{\Pi'}$. Now, $u[s']_\Pi \xrightarrow{*} u[l\sigma']_{\Pi''}$, by rewriting any additional occurrences of x in l that were not included in Π. To show that $l\sigma' \rightarrow r\sigma'$, we need to show that $p\sigma' \downarrow q\sigma'$. Let eq be a binary operator not appearing in any rule, and consider the derivation $eq(p\sigma,q\sigma) \underset{n-1}{\xrightarrow{*}} eq(w,w)$, known to exist for some w by virtue of the fact that $p\sigma \downarrow q\sigma$. We also have $p\sigma \rightarrow p\sigma'$ and $q\sigma \xrightarrow{*} q\sigma'$ by application of $s \rightarrow t$ at occurrences of s in the substitution parts. By induction on the second component, we have $eq(p\sigma',q\sigma') \downarrow eq(w,w)$, from which it follows (there being no rules for eq) that $p\sigma' \downarrow q\sigma'$. Thus, $u[s']_\Pi \xrightarrow{*} u[r\sigma]_{\Pi''} \downarrow v$. This is illustrated in Fig. 5.3.

Similarly, if π is inside some s, we have $u[s']_\Pi \xrightarrow{*} u[d\tau'] \downarrow v$, as shown in Fig. 5.4. Here τ' is like the substitution τ used to rewrite $s \rightarrow s'$, but maps the variable in g to $c[r\sigma]$ instead of to $c[l\sigma]$. The condition $p'\tau' \downarrow q'\tau'$, needed to show that $u' \xrightarrow{*} u[g\tau']_\Pi \rightarrow u[d\tau']_\Pi$, can be shown by induction on the first component, since $l\sigma$ is a proper subterm of s. □

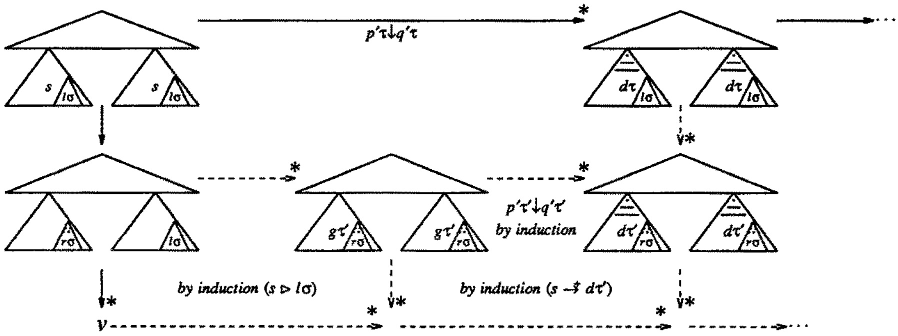

Figure 5.4. Variable overlap below (Lemma 2).

6. Conclusion

We have explored two different restrictions on critical pairs of conditional rewrite systems, namely shallow joinability and overlays only, and proved confluence results for systems meeting those restrictions. Our proofs show that, for conditional systems, the notions of confluence, local-confluence, and joinable critical pairs can not be neatly disentangled. In particular, the noetherian condition was needed to show that a system is locally confluent if all critical pairs are shallow joinable. We have also presented counter-examples which show that all our restrictions are necessary.

We have only given examples of systems not containing variables in the condition that do not also appear on the left-hand side. But in the programming context, at least, one would certainly like to allow rules such as:

$$(x \le y\ ,\ y \le z) \downarrow (tt\ ,\ tt): \quad x \le z \quad \to \quad tt,$$

where y is a "new" variable, or like:

$$fib(x) \downarrow \langle y\ ,\ z \rangle : \quad fib(s(x)) \quad \to \quad \langle y+z\ ,\ y \rangle,$$

where the right-hand side also has an occurrence of the new variables, y and z.

Even with new variables, the proofs of Theorems 2 and 4 should go through without difficulty. But, operationally, rewriting is more difficult now, since new variables in the conditions must be solved for. Thus, to rewrite an instance $l\sigma$ of a left-hand side, an interpreter must first find a *satisfying* substitution τ for the new variables in the condition $p \downarrow q$ such that $p\sigma\tau$ joins $q\sigma\tau$, and then replace $l\sigma$ by $r\sigma\tau$. One way to enumerate solutions (for decreasing and confluent systems) is via *narrowing*. Unfortunately, it is undecidable, in general, whether such a substitution exists.

Note, also, that with new variables on the right, a rule may non-trivially overlay itself. For example, a rewrites to $f(b)$ and $f(c)$ with the system

$$
\begin{array}{lrcl}
p(x) \downarrow tt: & a & \rightarrow & f(x) \\
 & p(b) & \rightarrow & tt \\
 & p(c) & \rightarrow & tt
\end{array}
$$

In general, a rule with new right-hand side variables can rewrite (in one step) to an infinite number of different terms (i.e. even finite systems may not be "locally finite").

To handle systems not meeting any of the known critical pair criteria, superposition (i.e. narrowing) of left-hand sides on conditions must also be considered. For example, given Example B, it might be nice to generate the following confluent unconditional system:

$$
\begin{array}{rcl}
c & \rightarrow & k(g(b)) \\
a & \rightarrow & b \\
h(x) & \rightarrow & k(x) \\
f(b) & \rightarrow & g(b)
\end{array}
$$

With this system, all unconditional equational consequences of Example B have "rewrite" proofs. This is an active area of research, the results of which may be particularly useful from the programming point of view.

REFERENCES

[Bergstra-Klop-86] Bergstra, J. A., and Klop, J. W. "Conditional rewrite rules: Confluency and termination", *JCSS* **32**, pp. 323-362, 1986.

[Dershowitz-Okada-Sivakumar-87] Dershowitz, N., Okada, M., and Sivakumar, G. "Canonical conditional rewrite systems", submitted.

[Dershowitz-Plaisted-87] Dershowitz, N., and Plaisted, D. A. "Equational programming". In *Machine Intelligence 11: The logic and acquisition of knowledge* (J. E. Hayes, D. Michie, and J. Richards, eds.), Clarendon Press, London, 1987.

[Fribourg-85] Fribourg, L. "SLOG: A logic programming language interpreter based on clausal superposition and rewriting". *Proceedings of the 1985 Symposium on Logic Programming*, Boston, MA (July 1985), pp. 172-184.

[Goguen-Meseguer-86] Goguen, J. A., and Meseguer, J. "EQLOG: Equality, types and generic modules for logic programming". In *Logic Programming: Functions, relations and equations* (D. DeGroot and G. Lindstrom, eds.), Prentice-Hall, Englewood Cliffs, NJ, pp. 295-363, 1986.

[Huet-80] Huet, G. "Confluent reductions: Abstract properties and applications to term rewriting systems", *J. ACM* **27**, pp. 797-821.

[Jouannaud-Waldmann-86] Jouannaud, J.-P., and Waldmann, B. "Reductive conditional term rewriting systems". *Proceedings of the Third IFIP Working Conference on Formal Description of Programming Concepts*, Ebberup, Denmark.

[Kaplan-87] Kaplan, S. "Simplifying conditional term rewriting systems: Unification, termination and confluence", *Journal of Symbolic Computation*, to appear.

[Zhang-Rémy-85] Zhang, H., and Rémy, J. L. "Contextual rewriting". *Proceedings First International Conference on Rewriting Techniques and Applications*, Dijon, France (May 1985), pp. 46-62 (available as Vol. 202, Lecture Notes in Computer Science, Springer, Berlin [September 1985]).