**AUTHOR**
Kapur, D
Narendran, P*

**SUBJECT**
rewriting systems

**NO.**
83CRD258

**DATE**
November 1983

**GE CLASS**
1

**NO. PAGES**
15

**SUMMARY**

The almost-confluence property of Thue systems was defined by Nivat and others to extend the Church-Rosser property so that size-preserving rules also play a role along with size-reducing rules in deciding the word problem of such Thue systems. We show that another related extension of the Church-Rosser property, called lex-confluence, in which like size-reducing rules, size-preserving rules are also treated as reductions, is orthogonal to the almost-confluence property. We discuss a single-rule almost-confluent system which is not equivalent to any finite noetherian (hence lex-confluent) Thue system, thus settling also an open question by Jantzen in a stronger way. We further show that the test for almost-confluence is PSPACE-complete which establishes an interesting complexity hierarchy for test of various properties of Thue systems since the check for confluence and Church-Rosserness is polynomial, whereas the pre-perfectness property is undecidable. We define the notion of a reduced almost-confluent system giving a normal form for such systems. Unlike for Church-Rosser and lex-confluent Thue systems, these normal forms are not unique; however, they are unique modulo an equivalence relation naturally defined on almost-confluent Thue systems. We discuss how the Knuth-Bendix completion procedure can be used to complete an arbitrary Thue system to obtain an equivalent almost-confluent Thue system whenever it exists.

*Dept. of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY

**KEY WORDS**
Thue systems, rewriting systems, almost-confluence, Church-Rosser property, lexicographic-confluence, PSPACE, confluence modulo equivalence, Knuth-Bendix procedure, word problem, normal forms
ALMOST-CONFLUENCE AND RELATED PROPERTIES OF THUE SYSTEMS*

D. Kapur† and P. Narendran

1. Introduction

The almost-confluence property of Thue systems [Book] has been defined to extend the Church-Rosser property (which means that no matter how a string is reduced, it always reduces to the same string) so that size-preserving rules also play a role in deciding the word problem of Thue systems. The almost-confluence property has a useful analog in Term Rewriting Systems, especially when they are being used for theorem-proving applications [Dershowitz; Musser; Musser and Kapur] as well as in canonical simplification in computer algebra systems [Buchberger and Loos; Kandri-Rody and Kapur]. A subset of axioms, there, can be treated as bidirectional simplification rules which either cannot be oriented without violating the well-foundedness property of a rewriting relation, or do not have to be considered as a part of rewriting since there already exists a canonical system for them (or in a weaker sense, there is a decision procedure for equivalence induced by these axioms) [see Kandri-Rody and Kapur]. This subset of axioms corresponds to the subset of size-preserving rules in Thue systems. The remaining axioms of the term rewriting system can be oriented and treated as a rewriting system which operates on top of this simplification (see also Lankford and Ballentyne, and Huet ['80] for related approaches).

Thue systems are a restriction on term rewriting systems where one does not have to consider variables or orientation of a subset of axioms as the orientation is built in the definition of Thue systems. Because of these properties and an additional property of associativity that strings satisfy, the study of Thue systems provides a useful insight for various properties of term rewriting systems.

In an earlier paper, we discussed conditions under which the Knuth-Bendix completion procedure [Knuth and Bendix] is guaranteed to terminate for Thue systems [Kapur and Narendran]. In that paper, we defined a property, called lex-confluence, of Thue system which is a generalization of the Church-Rosser property; instead of requiring that both sides of size-preserving rules reduce to the same normal form as in a Church-Rosser system, size-preserving rules in a lex-confluent system are oriented by choosing a suitable ordering on strings and are used as reductions. In studying the lex-confluence property of Thue systems, all axioms are thus oriented, analogous to term rewriting systems. The study of almost-confluence property allows size-preserving rules to be used in both directions. In this paper, we discuss the relationship between

---

* Some of the results reported in this paper will appear in P. Narendran’s doctoral dissertation at Rensselaer Polytechnic Institute, Troy, NY.

† Partially supported by NSF grant MCS-8211621.
the lex-confluence and almost-confluence properties of Thue systems and show that they are orthogonal properties. We exhibit an almost-confluent system having no finite noetherian (hence lex-confluent) Thue system equivalent to it, thus settling also an open question by Jantzen in a stronger way.

We also show that the test for almost-confluence is PSPACE-complete. This gives an interesting complexity hierarchy for the test of various properties of Thue systems: check for confluence and Church-Rosserness is polynomial (in fact, $O(T^3)$ as shown in [Kapur, Krishnamoorthy, McNaughton and Narendran]), check for almost-confluence is PSPACE-complete and check for pre-perfectness is undecidable [Narendran].

Some of the results obtained for lex-confluent Thue systems in [Kapur and Narendran] extend to almost-confluence systems in an obvious way. In particular, we define a normal form for almost-confluent Thue systems which is not unique in the sense of Church-Rosser and lex-confluent Thue systems but is unique modulo an equivalence relation naturally defined on almost-confluent Thue systems. Further, we also discuss how the Knuth-Bendix procedure can be used to complete an arbitrary Thue system to obtain an equivalent almost-confluent Thue system.

Finally, we discuss a generalization of the almost-confluence property on Thue systems in which an arbitrary subset of axioms that includes all size-preserving rules, is used to check for equivalence of words instead of just the size-preserving rules.

2. Definitions

Most of the definitions in this section are repeated from Book ["82] for the sake of completeness.

Let $\Sigma$ be a finite alphabet. $\Sigma^*$ is the set of all finite strings (words) over $\Sigma$, including the empty string $\lambda$ where $\lambda$ is not in $\Sigma$. The size function on strings, denoted by $|u|$, can be defined as usual: $|\lambda| = 0$, $|u a| = |u| + 1$, where $a$ is in $\Sigma$.

A Thue system $T$ is a binary relation on $\Sigma^*$. The Thue congruence generated by $T$ is the reflexive transitive closure $\rightarrow_T$ of the relation $\rightarrow_T$ which is defined as follows: for any $u$ and $v$ in $\Sigma^*$ such that $<u,v>$ is in $T$ or $<v,u>$ is in $T$, and any $x$, $y$ in $\Sigma^*$, $x uy \rightarrow_T x vy$. Two strings $w$ and $z$ are congruent mod $T$ if $w \rightarrow_T z$. Two Thue systems $T_1$ and $T_2$ are equivalent if and only if they generate the same congruence relation. Henceforth, we will omit the subscript $T$ whenever it is understood from the context.

Every element of a Thue system $T$ is called an equation of $T$. Some equations of $T$ can be oriented into rules depending upon the size of the sides of each equation. An equation $u = v$ is oriented into a rule $u \rightarrow v$ if $|u| > |v|$, otherwise, $v \rightarrow u$ when $|v| > |u|$; such a rule is called size-reducing or simply a reduction.

An equation $u = v$ in which $u$ and $v$ are of the same size, cannot be oriented based on the size of $u$ and $v$; it is written as $u \vdash v$. Such an equation is called a size-preserving rule. Later, we will discuss another way of orienting rules for the case when a total ordering is introduced on $\Sigma^*$. In that case, it would also be possible to uniquely orient equations whose two sides have strings of the same size.
Based on the above classification of the rules, a Thue system \( T \) can be partitioned into two components: (i) a subset of size-preserving rules, which will be called \( SP \), and (ii) the remaining subset of reductions, which will be called \( R \). Only the rules in the \( R \) subset of \( T \) will be used for reducing (rewriting) strings unless stated otherwise.

For any \( x \), if there are \( u \) and \( v \), as well as a rule \( l \to r \) in \( R \) of \( T \) such that \( x = u l v \), then \( x \to u r v \), read as \( x \) reduces to \( u r v \) using the rule \( l \to r \) of \( R \). The reflexive transitive closure of this relation is the reduction relation generated by \( R \) of \( T \) (also called the reduction relation generated by \( T \) on \( \Sigma^* \); this is denoted by \( \to^* \) whereas \( \to^+ \) stands for the transitive closure of \( \to \). Let \( \vdash^* \) stand for the congruence relation generated by the \( SP \) component of a Thue system \( T \). Caution: The relation \( \to^* \) is not \( (\to U \to)^* \), where \( \to \) is the inverse of the relation \( \to \); instead, \( \to^* = (\to U \to^*)^* \).

A Thue system \( T \) is called Church-Rosser if for each \( u, v \), such that \( u \to^* v \), there is a \( w \) such that \( u \to^* w \) and \( v \to^* w \).

\[
\begin{align*}
&\text{\begin{tikzpicture}[->,shorten >=1pt,auto,node distance=2cm,thick] \\
&\node (U) {\textcolor{red}{U}}; \\
&\node (V) [right of=U] {\textcolor{red}{V}}; \\
&\node (W) [below of=U] {\textcolor{red}{W}}; \\
&\node (Z) [below of=V] {\textcolor{red}{Z}}; \\
&\draw [->] (U) to (V); \\
&\draw [->] (U) to (W); \\
&\draw [->] (U) to (Z); \\
&\draw [->] (V) to (W); \\
&\draw [->] (V) to (Z); \\
&\draw [->] (W) to (Z); \\
&\end{tikzpicture}}
\end{align*}
\]

A Thue system \( T \) is called confluent if for each \( u, v, w \), such that \( u \to^* v, u \to^* w \), there is a \( z \) such that \( v \to^* z \) and \( w \to^* z \). Pictorially, we have

\[
\begin{align*}
&\text{\begin{tikzpicture}[->,shorten >=1pt,auto,node distance=2cm,thick] \\
&\node (U) {\textcolor{red}{U}}; \\
&\node (V) [above right of=U] {\textcolor{red}{V}}; \\
&\node (W) [below left of=U] {\textcolor{red}{W}}; \\
&\node (Z) [below right of=V] {\textcolor{red}{Z}}; \\
&\draw [->] (U) to (V); \\
&\draw [->] (U) to (W); \\
&\draw [->] (U) to (Z); \\
&\draw [->] (V) to (W); \\
&\draw [->] (V) to (Z); \\
&\draw [->] (W) to (Z); \\
&\end{tikzpicture}}
\end{align*}
\]

\( T \) is called almost-confluent if for each \( u, v \) such that \( u \to^* v \), there are \( w \) and \( z \) such that \( u \to^* w, v \to^* z \) and \( w \) and \( z \) are related by rules in \( SP \) component of \( T \), i.e. \( w \vdash^* z \). The following diagram depicts this property:

\[
\begin{align*}
&\text{\begin{tikzpicture}[->,shorten >=1pt,auto,node distance=2cm,thick] \\
&\node (U) {\textcolor{red}{U}}; \\
&\node (V) [right of=U] {\textcolor{red}{V}}; \\
&\node (W) [below of=U] {\textcolor{red}{W}}; \\
&\node (Z) [below of=V] {\textcolor{red}{Z}}; \\
&\draw [->] (U) to (V); \\
&\draw [->] (U) to (W); \\
&\draw [->] (U) to (Z); \\
&\draw [->] (V) to (W); \\
&\draw [->] (V) to (Z); \\
&\draw [->] (W) to (Z); \\
&\end{tikzpicture}}
\end{align*}
\]

\( T \) is called pre-perfect if for each \( u, v \) such that \( u \to^* v \), there is a \( w \) such that \( u \vdash^* w \) and \( v \vdash^* w \), where \( \vdash \) is \( \to \). Pictorially,
Proposition [Book]: If \( T \) is Church-Rosser, then \( T \) is confluent as well as almost-confluent. The following Thue system is confluent but neither Church-Rosser nor almost-confluent.

\[
T = \{ ab \rightarrow ef, ab \rightarrow c, ef \rightarrow d \}.
\]

For a Thue system \( T \), \( x \) is irreducible (mod \( T \)) if and only if \( x \) cannot be reduced further using the \( R \) component of \( T \); \( x \) is minimal if and only if \( x \) is one of the smallest strings in its congruence class induced by \( T \).

Let \( y \) be an irreducible string obtained by reducing \( x \) using rules of \( R \); \( y \) is also called a normal form of \( x \). To indicate the dependence of \( y \) on \( R \), we also denote \( y \) by normal_form\((x, R)\). If a string \( x \) has a unique normal form under \( R \), the normal form of \( x \) is also called its canonical form.

Let \( IRR(T) \) be the set of all irreducible strings of \( T \). Two equivalent Thue systems can have different \( IRR \) sets. It can be easily seen that equivalent Church-Rosser (lex-confluent) systems have the same \( IRR \) set; further for two equivalent Thue systems with the same \( IRR \) set, if one is Church-Rosser then so is the other [Kapur and Narendran]. For an almost-confluent system, the following holds:

**Lemma 2.1 [Berstel]:** For an almost-confluent system, a string \( w \) is irreducible if and only if it is minimal.

**Lemma 2.2 [Nivat]:** \( T \) is almost-confluent if and only if for irreducible \( x, y \) such that \( x \rightarrow^* y \) imply \( x \rightarrow^* y \).

Using Lemma 2.1, we get

**Lemma 2.3:** Let \( T_1 \) and \( T_2 \) be two equivalent almost-confluent systems. Then \( IRR(T_1) = IRR(T_2) \).

Strings \( w_1 \) and \( w_2 \) are said to be almost-joinable if and only if there exists \( z_1 \) and \( z_2 \) such that \( w_1 \rightarrow^* z_1, w_2 \rightarrow^* z_2 \) and \( z_1 \rightarrow^* z_2 \). A set of pairs of words is almost-joinable if and only if every pair in the set is almost-joinable.

### 3. Relating Almost-Confluence to Lex-Confluence and Confluence Mod -

In this section, we discuss how the almost-confluence property is related to two other similar properties — lex-confluence defined in Kapur and Narendran and confluence modulo - defined by Huet [’80].
3.1 Lex-Confluence

As stated above, the lex-confluence property of a Thue system is a generalization of the Church-Rosser property and appears similar to almost confluence as its definition given below suggests.

Let $<$ be a total ordering on strings in $\Sigma^*$ such that the following two properties hold:

(i) $|x| < |y| \Rightarrow x < y$, and

(ii) $x < y \Rightarrow$ for any $u, v, u x v < u y v$.

One such total ordering uses the lexicographic ordering induced by a total ordering on $\Sigma$, the alphabet, on strings of the same size. The total ordering $<$ on $\Sigma^*$ can be used to orient equations whose two sides are of the same size. So, given a Thue system $T$, equations in the size-preserving component can also be oriented using $<$. Every rule in $T$ is thus used for reduction, so $R$ and $T$ are the same for this case. The symbol $\rightarrow$ will be used to denote this reduction relation also as long as it is evident from the context that the whole $T$ is being used for reduction. In case of possible confusion, we will use $\rightarrow'$ to specify the reduction relation induced by $T$ to distinguish from the reduction relation $\rightarrow$ induced by $R$.

A Thue system $T$ is called lexicographically-confluent (with respect to $<$) if and only if for every $u, v,$ and $w$ such that $u \rightarrow^* v$ and $u \rightarrow^* w$, there is a $z$ such that $v \rightarrow^* z$ and $w \rightarrow^* z$. We abbreviate a lexicographically confluent system to a "lex-confluent system."

We now show that there exists an almost-confluent Thue system such that there is no finite noetherian system equivalent to it no matter what ordering on the strings we choose.

This Thue system is $T_1 = \{aba \leftrightarrow bab\}$. In $T_1$, we have

$abbab \rightarrow^* babba$.

Using this as the basis step ($i = 0, j = 0$), it can be shown by induction that

$$a^{i+1} b^{j+2} ab \rightarrow^* ba b^{i+2} a^{j+1} \quad i, j \geq 0,$$

Further, using $[bab] = \{aba, bab\}$, where for any $x$, $[x]$ stands for the equivalence class containing the word $x$, we can also show by induction that

$$[b^n ab] = \{b^{n-i} aba^i : 0 \leq i \leq n\}, n > 0,$$

Similarly,

$$[ba b^n] = \{a^j bab^{n-j} : 0 \leq j \leq n\}, n > 0, $$

Also observe that strings of the form in (1), i.e.,

$a^{i+1} b^{j+2} ab$ and $ba b^{i+2} a^{j+1}$
are such that they can be equated to other strings only because of substrings $b^k ab$ and $bab^k$, respectively. This is so because all other substrings of strings in (1) can be shown to have congruence classes with one element. Using these properties of $T_1$, we can show the following:

**Theorem 3.1:** There does not exist any finite noetherian system equivalent to $T_1$.

**Proof:** By contradiction; assume that there exists a finite noetherian system equivalent to $T_1$, then there is a reduced finite reduced noetherian system (which is obtained by reducing the left-hand side (lhs) and right-hand side (rhs) of rules in $T_1$ using other rules in $T_1$), call it $T_1'$, equivalent to $T_1$.

Since $[bab] = \{aba, bab\}$, any noetherian system should contain either $(aba \rightarrow' bab)$ or $(bab \rightarrow' aba)$. Without any loss of generality, assume the former is the case. Since the system is reduced, all rhs’s must be irreducible.

Let $L$ be the length of the longest left-hand side (lhs) of the desired noetherian system. Choose $i$ and $j$ to be $L$ in (1) above. That is,

$$a^{i+1} b^{j+2} ab \rightarrow^* bab^{j+2} a^{i+1}.$$

They should be joinable in $T_1'$. But the only substrings of these two strings that can possibly be reduced at the first step are strings of the form $b^k ab$ and $bab^k$, respectively, as all other substrings of strings in (1) can be shown to have congruence classes with one element.

That is, there must be rules of the form

$$b^i ab \rightarrow x \text{ and }$$

$$bab^j \rightarrow y$$

But $x$ and $y$ will be reducible by $(aba \rightarrow bab)$ as can be seen from (2) and (3). This contradicts the assumption that $T_1'$ is reduced. □

Jantzen had raised the question whether there is a finite Thue system with a decidable word problem which has no finite noetherian confluent system equivalent to it. We have proved a much stronger result that there is a single-rule, almost-confluent system which has no equivalent noetherian confluent system. We get the following as an immediate corollary of the above theorem.

**Theorem 3.2:** There does not exist any finite lex-confluent system equivalent to $T_1$.

3.2 Confluence Modulo

Book remarks that the almost-confluence property is similar to "confluence modulo an equivalence relation" defined by Huet [80].

According to Huet, $\rightarrow$ is confluent modulo an equivalence relation $\sim$, if for every $u, v, w$ and $z$ such that $u \sim v, u \rightarrow^* w, v \rightarrow^* z$, there are $x$ and $y$ such that $w \rightarrow^* x$ and $z \rightarrow^* y$ and $x \sim y$. Pictorially we have,
In contrast, the almost-confluence property can be pictorially depicted as:

An equivalent characterization of the almost-confluence property is stated in the following theorem whose proof is easy to see.

**Theorem 3.3:** A Thue system $T$ is almost-confluent if and only if for any $x, y$ such that $x \rightarrow^* y$, for any normal forms, $x'$ and $y'$ of $x$ and $y$, respectively, $x' \vdash^* y'$.

Note that any Thue system $T$ is trivially confluent modulo $\leftarrow^*$. The following theorem establishes that even if $\sim$ is $\vdash^*$, the congruence relation generated by the pairs in $SP$ component (size-preserving rules) of $T$, almost-confluence is equivalent to confluence modulo $\sim$.

**Theorem 3.4:** $T$ is almost-confluent if and only if $T$ is confluence modulo $\sim$, where $\sim = \vdash^*$.

**Proof:** The only if part is trivial, as $w \leftarrow^* z$ and almost-confluence ensures that there are $x$ and $y$ such that $w \rightarrow^* x$ and $z \rightarrow^* y$ and $x \vdash^* y$.

The if part is shown using Theorem 3.3. (This is similar to Lemma 2.6 in Huet ['80], given without proof.) Proof is by induction on the number of steps $n$ in $x \rightarrow^* y$.

**Basis:** $n = 0$, is immediate from confluence modulo $\vdash^*$.

**Inductive Step:** Assume for $n$, to show for $n+1$.

Let $x \rightarrow^* y_1 \rightarrow y$. There are three cases:

(a) $y_1 \vdash^* y$ : Normal forms of $y_1$ and $y$ are equivalent by $\vdash^*$ by confluence modulo $\vdash^*$. By the inductive hypothesis, normal forms of $y_1$ and $x$ are also $\vdash^*$-related. So normal forms of $x$ and $y$ are $\vdash^*$-related.

(b) $y_1 \rightarrow y$ : Normal form of $y_1$ via $y$ is $\vdash^*$ related to a normal form of $x$, so normal forms of $x$ and $y$ are $\vdash^*$-related.
(c) $y_1 \rightsquigarrow y$ : By confluence modulo $\vdash^*$, normal forms of $y_1$ and $y$ are $\vdash^*$-related. Using the inductive hypothesis, normal forms of $x$ and $y$ are $\vdash^*$-related. □

Note that the proofs of Theorems 3.3 and 3.4 do not depend on any particular characteristic of Thue systems, and that the theorems hold for any arbitrary reduction relation $\rightarrow$ and an equivalence relation $\sim$ such that $\vdash^* = (\rightarrow U \sim U \sim)^*$, using which the almost-confluence and confluence modulo $\sim$ properties can be defined accordingly (the equivalence relation $\sim$ need not be generated by a symmetric relation $\vdash$).

4. Test for Almost-Confluence

We show that the problem of testing the almost-confluence property of a Thue system is $P$-space complete. This establishes a hierarchy in terms of complexity on different properties of Thue systems as the tests for both the Church-Rosser and confluence properties are polynomial in complexity (in fact, $O(T^3)$ as shown in Kapur, Krishnamoorthy, McNaughton and Narendran), but the test for the almost-confluence property is $\text{PSPACE}$-complete, whereas the test for pre-perfectness is undecidable [Narendran].

Nivat and Benoist suggested a way to test for the almost-confluence property (which is the same as $\alpha$ and $\gamma$ conditions of Huet ['80] for the test for confluence modulo $\vdash^*$).

1. Superposition of two size-reducing rules $(b_i \rightarrow s_i)$ and $(b_j \rightarrow s_j)$:

   (a) If $b_i = uv$, $b_j = vw$ for some $u, v, w$ where $|ul|, |vl|, |wl| > 0$ (i.e., the two rules properly overlap), then for every such $u, v, w$, $(us_j, s_jw)$ must be almost-joinable. The string $uvw$ is called a superposition of the two rules, and $(us_j, s_jw)$ is called a critical pair.

   (b) If $b_i = ub_jw$ for some $u, w$ then for every such $u, w$, the critical pair $(s_i, us_jw)$ obtained from a superposition $b_i$ must be almost-joinable.

2. Superposition of a size-reducing rule $(b_i \rightarrow s_i)$ and a size-preserving rule $(l_j \vdash r_j)$ :

   (a) If $b_i = uv$, $l_j = vw$ for some $u, v, w$ where $|ul|, |vl|, |wl| > 0$, then for every such $u, v, w$, the critical pair $(s_jw, ur_j)$ from a superposition $uvw$ must be almost-joinable.

   (b) If $l_j = uv$, $b_i = vw$ for some $u, v, w$ where $|ul|, |vl|, |wl| > 0$, then for every such $u, v, w$, the critical pair $(us_j, l_jw)$ from a superposition $uvw$ must be almost-joinable.

   (c) If $b_i = ul_jw$ for some $u, w$ then for every such $u, w$, the critical pair $(s_i, ur_jw)$ from a superposition $b_i$ must be almost-joinable.

   (d) If $l_j = ub_jw$ for some $u, w$ then for every such $u, w$, the critical pair $(r_j, us_jw)$ from a superposition $l_j$ must be almost-joinable.
Since a size preserving rule is symmetric (i.e., it is applied in both directions), the above conditions are also checked for $r_j$ and $b_i$. It is possible to state the above conditions succinctly by combining the cases such that one rule is size-reducing whereas the other rule could either be size-reducing or size-preserving.

4.1 Complexity Analysis

**Lemma 4.1:** Given a Thue system $T$, testing whether two words $x$ and $y$ are almost-joinable can be done in space polynomial in sizes of $x$ and $y$.

**Proof:** We can easily construct a non-deterministic linear-bounded automata (LBA) which will do this for us. Assume the input is $#x^*y@$ where # and @ are the left and right endmarkers and * any new symbol to separate $x$ from $y$. Now, the NLBA correctly ‘guesses’ the sequence of rules to be applied to $x$ and $y$ until the tape contents are of the form $#w^*w@$. No extra space is needed since we never use an expanding rule. □

**Theorem 4.2:** Almost-confluence can be tested in space polynomial of the size of the longest word on the lhs of a size-reducing rule or on either side of a size-preserving rule.

**Proof:** Since the test for almost-confluence needs to (i) generate all critical pairs which can be done in polynomial time and space, and (ii) check whether two strings are almost-joinable, we get from the previous lemma that almost-confluence can be tested in polynomial space. And, the size of the longest critical pair is proportional to the size of the longest word on the lhs of a size-reducing rule or either side of a size-preserving rule. □

**Lemma 4.3:** Testing for almost-confluence is PSPACE-hard.

**Proof:** Here we make use of the result that the uniform word problem for balanced Thue systems (systems in which all rules are size-preserving) is PSPACE-hard [Book et al.]. Given a balanced Thue system $T$ and words $x$ and $y$, we construct the following Thue system $T'$ comprising rules of $T$ and

$\#^n \rightarrow x$

$\#^n \rightarrow y$,

where $n = \max(|x|, |y|)$ and # and @ are new symbols. Now it can easily be seen that $T'$ is almost-confluent if and only if $x$ and $y$ are congruent modulo $T$, i.e. $x \equiv y \pmod{T}$. □

Thus, we have

**Theorem 4.4:** Testing for almost-confluence is PSPACE-complete.

5. Reduced Almost-Confluent System

Given an almost-confluent Thue system, it is possible to obtain a reduced almost-confluent Thue system from it by getting rid of redundancies. Before giving the algorithm for transforming
an almost-confluent system to a reduced almost-confluent system, we prove the following lemma which is useful in establishing the correctness of the transformations to get a reduced system.

**Lemma 5.1:** Let \( T_1 \) be a Thue system and \( T_2 \) an equivalent almost-confluent system. Let \( T' = \{ (x, y) \mid (x, y) \text{ in } T_2 \text{ but not in } T_1 \} \) (i.e., \( T' = T_2 - T_1 \)). Then, \( T_1 \) is almost-confluent if and only if every pair in \( T' \) is almost-joinable in \( T_1 \).

**Proof:** The 'only if' part is trivial.

For the 'if' part, assume the contrary. Then, by Lemma 2.2, there exist \( x, y \) in \( IRR(T_1) \) such that \( x \rightarrow^* T_1 y \) but not \( x \vdash^* T_1 y \) in \( T_1 \). We first prove the following lemma:

**Lemma 5.2:** \( IRR(T_1) = IRR(T_2) \)

**Proof:** (i) \( IRR(T_1) \subseteq IRR(T_2) \).

Let \( x \) be in \( IRR(T_1) \) but not in \( IRR(T_2) \). Then there exists a rule \( (b \rightarrow s) \) in \( T_2 \) (but not in \( T_1 \)) that reduces \( x \). Since \( (b, s) \) is almost-joinable by our assumption, \( b \) must be reducible in \( T_1 \). So, \( x \) is reducible in \( T_1 \) which is a contradiction.

(ii) \( IRR(T_2) \subseteq IRR(T_1) \).

If \( x \) is irreducible in \( T_2 \) and reducible in \( T_1 \), \( x \) cannot be minimal, which using Lemma 2.1, implies that \( x \) is reducible in \( T_2 \), which is a contradiction. \( \square \)

Using Lemma 5.2, we have \( x \) and \( y \) are irreducible in \( T_2 \), so it must be that \( x \vdash^* T_2 y \) in \( T_2 \). Further, every size-preserving rule used in this transformation sequence must have both sides irreducible (or else minimality of \( x \) or \( y \) in \( T_2 \) will not hold). The application of every size-preserving rule in \( T' \) can be replaced by a size-preserving transformation in \( T_1 \); this is so because for \( (l \vdash r) \) be in \( T' \), if \( l, r \) are in \( IRR(T_1) \), then \( l \vdash^* T_1 r \). Thus \( x \vdash^* T_1 y \) modulo \( T_1 \), which is contrary to the assumption. \( \square \)

Here is an algorithm for obtaining a reduced almost-confluent system from an almost-confluent system:

1. (Replacement) For every size-reducing rule \( (b \rightarrow s) \), if \( s \) is reducible, then replace the rule by \( b \rightarrow s' \), where where \( s' \) is a normal form of \( s \).

2. (Removal) (i) For every size-reducing rule \( (b \rightarrow s) \), if \( b \) is reducible by any other rule in the system, then delete the rule \( b \rightarrow s \).

   (ii) For every size-preserving rule \( (l \vdash r) \), if either \( l \) or \( r \) is reducible, then delete the rule \( l \vdash r \).

**Claim 1:** Step 1 preserves equivalence and almost-confluence.

**Proof:** The rule \( (b \rightarrow s) \) can never be applied in the reduction \( s \rightarrow^* s' \). Thus \( (b, s) \) is almost-joinable in the system \( (T - \{ (b, s) \}) U \{ (b, s') \} \). Using Lemma 5.1, we get almost-confluence.

**Claim 2:** Step 2 preserves equivalence and almost-confluence.
Proof: Case of size-reducing rules: Let $b \rightarrow^+ b'$ by some rule other than $(b \rightarrow s)$. Now $b'$ and $s$ must be almost-joinable; since both are shorter than $b$, they are almost-joinable without using $(b \rightarrow s)$. Thus $(b, s)$ is almost-joinable in $T \setminus \{(b, s)\}$. Again, using Lemma 5.1, we get almost-confluence. The case of size-preserving rules is similar. □

5.1 A Normal Form for Almost-Confluent Systems

Using the above reduction steps, we can obtain an equivalent reduced almost-confluent system from every almost-confluent system. The following theorem tells us some more about the properties of reduced systems.

Theorem 5.3: Let $T_1$ and $T_2$ be two equivalent reduced almost-confluent systems.

(a) $SP(T_1)$ is equivalent to $SP(T_2)$, i.e., the congruent relations generated by the size-preserving components of $T_1$ and $T_2$ are the same.

(b) For every rule $(b \rightarrow s)$ in $R(T_1)$ there exists a rule $(b \rightarrow s')$ in $R(T_2)$ such that $s \vdash \ast s'$ and vice versa.

Proof: (a) Assume there exists a rule $(x \vdash y)$ in $T_1$ such that not $x \vdash \ast y$ in $T_2$. Since $T_2$ is almost-confluent, this implies that $x$ and $y$ are not irreducible and hence not minimal in $T_2$. Thus $x$ and $y$ are reducible in $T_1$ also (by Lemma 5.2) which contradicts the fact that $T_1$ is reduced.

(b) $b$ and $s$ are almost-joinable in $T_2$. Since all rhs’s in a reduced system are irreducible, $s$ is irreducible both in $T_1$ and $T_2$. Since no proper substring of $b$ can be reducible, there must be a rule $(b \rightarrow t)$, $t$ irreducible. Therefore $s$ and $t$ are equivalent and irreducible and this implies $s \vdash \ast t$ (by Lemma 2.2). □

6. The Knuth-Bendix Completion Procedure for Almost-Confluence

Given a Thue system $T$ which is not almost-confluent, it is possible to generate a reduced almost-confluent system equivalent to $T$, if such a system exists, from $T$ using the Knuth-Bendix completion procedure. In the test for almost-confluence, if for any pair of rules, the conditions are not met, i.e., the two strings generated from the superposition do not have the same normal form, then we modify the system by adding a rule which ensures the same normal form for that critical pair; in this way, we keep adding new rules whenever the need arises until the almost-confluence test is met.

Knuth-Bendix Procedure ($T$):

\[ i := 0 \]
\[ T_0 := \text{Normalize}(T) \]
while not almost_confluent($T_i$) do
\[ T_{i+1} := \text{Normalize}(T_i, CP(T_i)) \]
\[ i := i + 1 \]
end
output($T$)
Normalize($T$):
unmark all rules in $T$;
while $T$ has an unmarked rule $<l, r>$ do
$$T' := T - \{<l, r>\},$$
$$<l', r'> := <\text{normal form}(l, T'), \text{normal form}(r, T')>$$
if almost_joinable ($l', r', T'$) then $T := T'$
else if $<l, r> \neq <l', r'>$ then $\{T := T' U \{<l', r'>\}$$
mark $<l', r'>$
end
else mark $<l, r>$
return $T$

In the above procedures, the procedure CP generates all nontrivial critical pairs (which do not reduce to normal forms equivalent by $SP(T)$); the procedure almost_confluent tests for the almost-confluence of $T$. In an implementation, the two procedures are combined so that whenever the almost-confluence test fails, the critical pairs are generated by almost_confluent procedure itself. The procedure normal_form generates a normal form of $x$ using the size-reducing rules in $T$, whereas almost_joinable checks whether two strings are equivalent using the size-preserving rules of $T$. Note that words in nontrivial critical pairs after normalization may be of the same size; so, they are added to the size-preserving component of $T$ and are subsequently used to check for almost-joinability condition; they cannot be used in the reduction process. The above completion procedure to generate almost-confluent systems is different from other uses of the Knuth-Bendix completion procedure discussed in the literature, as in this case, the simplification theory generated by size-preserving rules is also being extended; so some of the new rules being generated are used as reduction while others are used as simplifications.

The procedure discussed above is not necessarily efficient as critical pairs among various rules are being checked for again and again; an efficient implementation can be designed based on a version of the procedure given in Huet [81].

We can extend the results discussed in [Kapur and Narendran] for lex-confluent Thue systems to almost-confluent systems. Using the techniques and proofs developed there, we can show that for a given Thue system $T$, if there exists a finite almost-confluent Thue system $T'$, then the Knuth-Bendix completion procedure terminates with a reduced almost-confluent Thue system $T''$ equivalent to $T$. We will not reproduce the proofs here but an interested reader should refer to [Kapur and Narendran] for details.

Examples:
1. $T = \{a \rightarrow b, bab \rightarrow b\}$.

After the first iteration of the completion procedure, three rules $baa \rightarrow b, aab \rightarrow b$, and $bbb \rightarrow b$ are added. In the next iteration, three rules $aaa \rightarrow b, bba \rightarrow b, abb \rightarrow b$ are added. Subsequently, the rule $aba \rightarrow b$ is added which makes the final system almost-confluent. The result is:
\{a \rightarrow b, aab \rightarrow b, aba \rightarrow b, abb \rightarrow b, baa \rightarrow b, bab \rightarrow b, bba \rightarrow b, bbb \rightarrow b\}

2. \(T = \{baa \rightarrow aab, bab \rightarrow a, bb \rightarrow a\}\)

First Iteration: \(aa \rightarrow a, ab \rightarrow ba\) added and \(baa \rightarrow aab\) deleted.

Second Iteration: \(aba \rightarrow ab\) added.

After this, the system

\{aa \rightarrow a, bb \rightarrow a, bab \rightarrow a, aba \rightarrow ab, ab \rightarrow ba\}

is almost-confluent.

7. **Generalization of almost-confluence**

Let us repeat the definition of almost-confluence:

![Diagram]

Only the size-preserving rules are used for checking equivalence of strings \(w\) and \(z\) obtained after reduction. It is possible to weaken this condition and instead, only require that any decidable equivalence relation hold between \(w\) and \(z\).

Let \(T\) be a Thue system and \(T'\) a subset of \(T\) such that all size-preserving rules of \(T\) are in \(T'\). \(T\) is called confluent mod \(T'\) if for any \(x, y\) such that \(x \rightarrow^* y\), there exists \(w\) and \(z\) such that \(x \rightarrow^*_{(T-T')} w\) and \(y \rightarrow^*_{(T-T')} z\) and \(w \rightarrow^*_{T'} z\). Pictorially, we have,

![Diagram with T and T']

Using Huet's results (Lemma 2.7 in particular) and results in Section 3.2, it can be shown that \(T\) is confluent mod \(T'\) if and only if the following conditions hold:

(i) for all \(x, y\) and \(z, x \rightarrow^*_{(T-T')} y\) and \(x \rightarrow^*_{(T-T')} z\), there exist \(w_1\) and \(w_2\) such that \(y \rightarrow^*_{(T-T')} w_1, z \rightarrow^*_{(T-T')} w_2\), and \(w_1 \rightarrow^*_{T'} w_2\).
(ii) for all \(x, y\) and \(z\), \(x \rightarrow^* y\) and \(x \rightarrow_{(T-T)} z\), there exist \(w_1\) and \(w_2\) such that \(y \rightarrow_{(T-T)} w_1\), \(z \rightarrow_{(T-T)} w_2\), and \(w_1 \rightarrow^* w_2\).

It should be obvious that almost-confluence is an instance of the above definition when \(T\) contains only the size-preserving rules.

Under certain conditions on \(T\), it is possible to further localize the (ii) property. Extending Huet’s results for confluence modulo \(\sim\), where \(\sim\) is the reflexive transitive closure of a symmetric relation, it can be shown that if

\[\rightarrow_{(T-T)} \cdot \rightarrow^*\] is noetherian, then the property (ii) above is equivalent to:

(ii-a) for all \(x, y\) and \(z\), \(x \rightarrow^* y\) and \(x \rightarrow_{(T-T)} z\), there exist \(w_1\) and \(w_2\) such that \(y \rightarrow_{(T-T)} w_1\), \(z \rightarrow_{(T-T)} w_2\), and \(w_1 \rightarrow^* w_2\),

(ii-b) for all \(x, y\) and \(z\), \(x \rightarrow y\) and \(x \rightarrow_{(T-T)} z\), there exist \(w_1\) and \(w_2\) such that \(y \rightarrow_{(T-T)} w_1\), \(z \rightarrow_{(T-T)} w_2\), and \(w_1 \rightarrow^* w_2\) and

(ii-c) for all \(x, y\) and \(z\), \(x \rightarrow y\) and \(x \rightarrow_{(T-T)} z\), there exist \(w_1\) and \(w_2\) such that \(y \rightarrow_{(T-T)} w_1\), \(z \rightarrow_{(T-T)} w_2\), and \(w_1 \rightarrow^* w_2\).

From the above, we have:

**Theorem 7.1:** For a Thue system \(T\) such that \(\rightarrow_{(T-T)} \cdot \rightarrow^*\) is noetherian and \(\rightarrow^*\) is decidable, confluence mod \(T\) of \(T\) is decidable.

**REFERENCES**


