# CoLL: A Confluence Tool for Left-Linear Term Rewrite Systems

Kiraku Shintani and Nao Hirokawa(✉)

School of Information Science, JAIST, Nomi, Japan
{s1310032,hirokawa}@jaist.ac.jp

**Abstract.** We present a confluence tool for left-linear term rewrite systems. The tool proves confluence by using Hindley's commutation theorem together with three commutation criteria, including Church-Rosser modulo associative and/or commutative theories. Despite a small number of its techniques, experiments show that the tool is comparable to recent powerful confluence tools.

## 1 Introduction

In this paper we present the new confluence tool CoLL for left-linear term rewrite systems (TRSs). The tool has two distinctive features. One is use of Jouannaud and Kirchner's theorem for the Church-Rosser modulo property. Our tool supports rewriting modulo associativity *and/or* commutativity rules. Another notable feature is that confluence is proved by commutation criteria only. By using Hindley's Commutation Theorem [1] confluence is proved via commutation of subsystems of an input TRS. In addition to them, CoLL implements a simple transformation technique that eliminates redundant rewrite rules.

The remaining part of the paper is organized as follows: In Sect. 3 we discuss how to use the Church-Rosser modulo theorem for associativity and/or commutativity rules. Commutation criteria and decomposition techniques supported in the tool are described in Sect. 4. In Sect. 5 we report experimental results to assess effectiveness of the presented techniques and the tool. The final section describes related work and concluding remarks. The tool is available at:

http://www.jaist.ac.jp/project/saigawa/coll/

## 2 Preliminaries

We assume familiarity of term rewriting and unification theory [2]. We recall here only some notions for rewriting and rewriting modulo. We consider terms built from a signature $\mathcal{F}$ and a set $\mathcal{V}$ of variables. We write $s \rhd t$ if $t$ is a proper subterm of $s$.

Supported by JSPS KAKENHI Grant Number 25730004 and Core to Core Program.

A.P. Felty and A. Middeldorp (Eds.): CADE-25, LNAI 9195, pp. 127–136, 2015.
DOI: 10.1007/978-3-319-21401-6_8

*Commutation.* Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs. We say that $\mathcal{R}$ and $\mathcal{S}$ commute if ${}_{\mathcal{R}}^{*}\!\!\leftarrow \cdot \rightarrow_{\mathcal{S}}^{*} \subseteq \rightarrow_{\mathcal{S}}^{*} \cdot {}_{\mathcal{R}}^{*}\!\!\leftarrow$. Confluence of $\mathcal{R}$ is equivalent to *self-commutation* of $\mathcal{R}$, i.e., commutation of $\mathcal{R}$ and $\mathcal{R}$. The relation $\rightarrow_{\mathcal{S}}^{*} \cdot \rightarrow_{\mathcal{R}} \cdot \rightarrow_{\mathcal{S}}^{*}$ is called the *relative step* of $\mathcal{R}$ over $\mathcal{S}$, and denoted by $\rightarrow_{\mathcal{R}/\mathcal{S}}$. We say that $\mathcal{R}/\mathcal{S}$ is terminating if $\rightarrow_{\mathcal{R}/\mathcal{S}}$ is terminating.

*Multi-steps.* The *multi-step* $\mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}}$ of a TRS $\mathcal{R}$ is inductively defined on terms as follows:

1. $x \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}} x$ for all $x \in \mathcal{V}$,
2. $f(s_1, \ldots, s_n) \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}} f(t_1, \ldots, t_n)$ if $s_i \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}} t_i$ for all $1 \leqslant i \leqslant n$, and
3. $\ell\sigma \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}} r\tau$ if $\ell \rightarrow r \in \mathcal{R}$ and $\sigma$ and $\tau$ are substitutions that $x\sigma \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{R}} x\tau$ for all variables $x$.

*Rewriting Modulo.* Let $\mathcal{R}$ and $\mathcal{E}$ be TRSs. The rewrite step $\rightarrow_{\mathcal{R},\mathcal{E}}$ of $\mathcal{R}$ *modulo theory* $\mathcal{E}$ is defined as follows: $s \rightarrow_{\mathcal{R},\mathcal{E}} t$ if If $s|_p \leftrightarrow_{\mathcal{E}}^{*} \ell\sigma$ and $t = s[r\sigma]_p$ for some position $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(s)$, rule $\ell \rightarrow r \in \mathcal{R}$, and substitution $\sigma$. The relation $\rightarrow_{\mathcal{R},\mathcal{E}}$ is *Church-Rosser modulo* $\mathcal{E}$, denoted by $\mathsf{CR}(\mathcal{R}, \mathcal{E})$, if $\leftrightarrow_{\mathcal{R}\cup\mathcal{E}}^{*} \subseteq \rightarrow_{\mathcal{R},\mathcal{E}}^{*} \cdot \leftrightarrow_{\mathcal{E}}^{*} \cdot {}_{\mathcal{R},\mathcal{E}}^{*}\!\!\leftarrow$. Let $\mathcal{F}_{\mathsf{A}}$, $\mathcal{F}_{\mathsf{C}}$, and $\mathcal{F}_{\mathsf{AC}}$ be pairwise disjoint sets of binary function symbols. We define the three theories $\mathsf{A}$ (associativity), $\mathsf{C}$ (commutativity), and $\mathsf{AC}$ as:

$$\begin{aligned}
\mathsf{A} &= \{f(f(x,y),z) \rightarrow f(x,f(y,z)),\ f(x,f(y,z)) \rightarrow f(f(x,y),z) \mid f \in \mathcal{F}_{\mathsf{A}}\} \\
\mathsf{C} &= \{f(x,y) \rightarrow f(y,x) \mid f \in \mathcal{F}_{\mathsf{C}}\} \\
\mathsf{AC} &= \{f(f(x,y),z) \rightarrow f(x,f(y,z)),\ f(x,y) \rightarrow f(y,x) \mid f \in \mathcal{F}_{\mathsf{AC}}\}
\end{aligned}$$

*Critical Pairs.* Conditions for confluence are often based on the notion of critical pairs. We denote by $\mathcal{U}_{\mathcal{E}}(s \approx t)$ a fixed complete set of $\mathcal{E}$-unifiers for terms $s$ and $t$. Let $\ell_1 \rightarrow r_1$ be a rule in a TRS $\mathcal{R}$ and $\ell_2 \rightarrow r_2$ a variant of a rule in a TRS $\mathcal{S}$ with $\mathcal{V}\mathsf{ar}(\ell_1) \cap \mathcal{V}\mathsf{ar}(\ell_2) = \varnothing$. When $p \in \mathcal{P}\mathsf{os}_{\mathcal{F}}(\ell_2)$ and $\sigma \in \mathcal{U}_{\mathcal{E}}(\ell_1 \approx \ell_2|_p)$, the pair $(\ell_2\sigma[r_1\sigma]_p, r_2\sigma)$ is called an *$\mathcal{E}$-extended critical pair* (or simply *$\mathcal{E}$-critical pair*) of $\mathcal{R}$ on $\mathcal{S}$, and written $\ell_2\sigma[r_1\sigma]_p\ {}_{\mathcal{R},\mathcal{E}}\!\!\leftarrow\!\rtimes\!\rightarrow_{\mathcal{S}} r_2\sigma$.

## 3 Confluence via Church-Rosser Modulo

In this section we explain how the next theorem by Jouannaud and Kirchner [3] is used for confluence analysis. Especially, we discuss how to deal with associativity and/or commutativity rules.

**Theorem 1.** *Let $\mathcal{R}$ and $\mathcal{E}$ be TRSs that $\mathcal{R}/\mathcal{E}$ is terminating and $\rhd \cdot \leftrightarrow_{\mathcal{E}}^{*}$ is well-founded. Then,* $\mathsf{CR}(\mathcal{R}, \mathcal{E})$ *iff* ${}_{\mathcal{R},\mathcal{E}}\!\!\leftarrow\!\rtimes\!\rightarrow_{\mathcal{R}\cup\mathcal{E}\cup\mathcal{E}^{-1}} \subseteq \rightarrow_{\mathcal{R},\mathcal{E}}^{*} \cdot \leftrightarrow_{\mathcal{E}}^{*} \cdot {}_{\mathcal{R},\mathcal{E}}^{*}\!\!\leftarrow$. □

We use the next left-linear TRS $\mathcal{R}_1$ to illustrate problems that arise when employing Theorem 1:

$$1\colon \mathsf{0} + x \rightarrow x \qquad 2\colon x + (y + z) \rightarrow (x + y) + z \qquad 3\colon (x + y) + z \rightarrow x + (y + z)$$

Let $\mathcal{F}_{\mathsf{A}} = \{+\}$. We may assume $\mathsf{A} = \{2, 3\}$. An idea here is proving $\mathsf{CR}(\{1\}, \mathsf{A})$ to conclude confluence of $\mathcal{R}_1$. The next trivial lemma validates this idea. We call a TRS $\mathcal{E}$ *reversible* if $\rightarrow_{\mathcal{E}} \subseteq {}_{\mathcal{E}}^{*}\!\!\leftarrow$ holds.

**Lemma 2.** *Suppose $\mathcal{E}$ is reversible. If $\mathsf{CR}(\mathcal{R},\mathcal{E})$ then $\mathcal{R} \cup \mathcal{E}$ is confluent.* □

Reversibility of $\mathsf{A}$ and well-foundedness of $\rhd \cdot \leftrightarrow^*_{\{2,3\}}$ are trivial. Termination of $\{1\}/\mathsf{A}$ can be shown by AC-RPO [4]. Therefore, it remains to test joinability of extended critical pairs to apply Theorem 1.

### 3.1 Associative Unification

How to compute $\mathsf{A}$-critical pairs? Plotkin [5] introduced a procedure that enumerates a minimal complete set of $\mathsf{A}$-unifiers. It is well-known that a minimal complete set need not to be finite, and thus the procedure may not terminate. In fact there is a one-rule TRS that admits infinitely many $\mathsf{A}$-critical pairs. Probably this is one of main reasons that existing confluence tools do not support Theorem 1 for associativity theory. However, as observed in [6], a minimal complete set resulting from the procedure is finite whenever an input equality is a pair of linear terms that share no variables. Therefore, for every left-linear TRS one can safely use Plotkin's procedure to compute their $\mathsf{A}$-critical pairs.

We present a simple variant of Plotkin's procedure [5,7] specialized for our setting. Let $S$ and $T$ be sets of substitutions. We abbreviate the set $\{\sigma\tau \mid \sigma \in S \text{ and } \tau \in T\}$ to $ST$. Given a term $t$, we write $t{\downarrow_{\mathsf{A}'}}$ for the normal form of $t$ with respect to $\mathsf{A}'$. Here $\mathsf{A}'$ stands for the confluent and terminating TRS $\{f(f(x,y),z) \to f(x,f(y,z)) \mid f \in \mathcal{F}_{\mathsf{A}}\}$.

**Definition 3.** *Let $s$ and $t$ be terms. The function $\langle s \approx t\rangle$ is inductively defined as follows:*

$$\langle s \approx t\rangle = \begin{cases} \{\{s \mapsto t\}\} & \text{if } s \in \mathcal{V} \\ \{\{t \mapsto s\}\} & \text{if } s \notin \mathcal{V} \text{ and } t \in \mathcal{V} \\ A_1 \cdots A_n \cup A_{s,t} \cup A_{t,s} & \text{if } s = f(s_1,\ldots,s_n) \text{ and } t = f(t_1,\ldots,t_n) \\ \varnothing & \text{otherwise} \end{cases}$$

*where,*

$$A_i = \langle s_i \approx t_i\rangle, \qquad A_{s,t} = \begin{cases} \{\{s_1 \mapsto f(t_1,s_1)\}\}\langle s \approx t_2\rangle & \text{if } (*) \\ \varnothing & \text{otherwise} \end{cases}$$

*and $(*)$ stands for $s = f(s_1,s_2)$, $t = f(t_1,t_2)$, $f \in \mathcal{F}_{\mathsf{A}}$, and $s_1 \in \mathcal{V}$.*

**Theorem 4.** *Let $s$ and $t$ be linear terms with $\mathcal{V}\mathsf{ar}(s) \cap \mathcal{V}\mathsf{ar}(t) = \varnothing$. The set $\langle s{\downarrow_{\mathsf{A}'}} \approx t{\downarrow_{\mathsf{A}'}}\rangle$ is a finite complete set of their $\mathsf{A}$-unifiers.* □

We illustrate the use of the theorem. Let $s = \mathsf{0} + x$, $t = (x' + y') + z'$, and $\mathcal{F}_{\mathsf{A}} = \{+\}$. A complete set of $\mathsf{A}$-unifiers for the terms is computed as follows:

$$\begin{aligned} \langle s{\downarrow_{\mathsf{A}'}} \approx t{\downarrow_{\mathsf{A}'}}\rangle &= \langle \mathsf{0} + x \approx x' + (y' + z')\rangle \\ &= \big(\langle \mathsf{0} \approx x'\rangle\langle x \approx y' + z'\rangle\big) \cup \big(\{\{x' \mapsto \mathsf{0} + x'\}\}\langle x' + (y' + z') \approx x\rangle\big) \\ &= \big\{\{x' \mapsto \mathsf{0}, x \mapsto y' + z'\},\ \{x' \mapsto \mathsf{0} + x', x \mapsto x' + (y' + z')\}\big\} \end{aligned}$$

This set induces the $\mathsf{A}$-critical pairs:

$$\begin{array}{rcl} y' + z' & {}_{\{1\},\mathsf{A}}\!\leftarrow \rtimes \rightarrow_{\{3\}} & \mathsf{0} + (y' + z') \\ x' + (y' + z') & {}_{\{1\},\mathsf{A}}\!\leftarrow \rtimes \rightarrow_{\{3\}} & (\mathsf{0} + x') + (y' + z') \end{array}$$

Both of the right-hand sides reduce to the corresponding left-hand sides by the rewriting modulo step $\rightarrow_{\{1\},\mathsf{A}}$. What about the other critical pairs?

### 3.2 Coherence

Consider the $\mathsf{A}$-critical pair:

$$x + z \quad {}_{\{1\},\mathsf{A}}\!\leftarrow\!\rtimes\!\rightarrow_{\{2\}} \quad (x + \mathsf{0}) + z$$

Contrary to our intention, $(x + \mathsf{0}) + z \rightarrow_{\{1\},\mathsf{A}} x + z$ does not hold, and thus $\mathsf{CR}(\{1\}, \mathsf{A})$ is refuted by Theorem 1. This undesired incapability of rewriting modulo is known as the coherence problem [3,8].

**Definition 5.** *A pair $(\mathcal{R}, \mathcal{E})$ is* strongly coherent *if* $\leftrightarrow_{\mathcal{E}}^{*} \cdot \rightarrow_{\mathcal{R},\mathcal{E}} \subseteq \rightarrow_{\mathcal{R},\mathcal{E}} \cdot \leftrightarrow_{\mathcal{E}}^{*}$.

**Lemma 6.** *Suppose $\mathcal{E}$ is reversible and $(\mathcal{R}, \mathcal{E})$ is strongly coherent. If $\mathsf{CR}(\mathcal{R}, \mathcal{E})$ then $\mathcal{R} \cup \mathcal{E}$ is confluent, and vice versa.* □

While the strong coherence property always holds for rewriting modulo $\mathsf{C}$, rewriting modulo $\mathsf{A}$ and $\mathsf{AC}$ rarely satisfy the property. This can be overcome by *extending* a rewrite system. Since an extension for $\mathsf{AC}$ is known [3,8], here we consider an $\mathsf{A}$-extension of a TRS.

**Definition 7.** *Let $\mathcal{R}$ be a TRS. The* $\mathsf{A}$-extended *TRS $\mathsf{Ext}_{\mathsf{A}}(\mathcal{R})$ consists of*

$$\begin{array}{lll} \ell \rightarrow r & f(\ell, x) \rightarrow f(r, x) & f(x, f(\ell, y)) \rightarrow f(x, f(r, y)) \\ & f(x, \ell) \rightarrow f(x, r) & \end{array}$$

*for all rules $\ell \rightarrow r \in \mathcal{R}$ with $f = \mathrm{root}(\ell) \in \mathcal{F}_A$. Here $x$ and $y$ are fresh variables not in $\ell$.*

**Lemma 8.** *The pair $(\mathsf{Ext}_{\mathsf{A}}(\mathcal{R}), \mathsf{A})$ is strongly coherent and $\rightarrow_{\mathsf{Ext}_{\mathsf{A}}(\mathcal{R})} = \rightarrow_{\mathcal{R}}$.*

*Proof.* From the inclusion $\rightarrow_{\mathsf{A}} \cdot \rightarrow_{\mathsf{Ext}_{\mathsf{A}}(\mathcal{R}),\mathsf{A}} \subseteq \rightarrow_{\mathsf{Ext}_{\mathsf{A}}(\mathcal{R}),\mathsf{A}} \cdot \rightarrow_{\mathsf{A}}^{*}$ the first claim follows. Since $\rightarrow_{\mathcal{R}}$ is closed under contexts, the second claim is trivial. □

The TRS $\mathsf{Ext}_{\mathsf{A}}(\{1\})$ consists of the four rules:

$$\begin{array}{cc} \mathsf{0} + x \rightarrow x & w + (\mathsf{0} + x) \rightarrow w + x \\ (\mathsf{0} + x) + y \rightarrow x + y & w + ((\mathsf{0} + x) + y) \rightarrow w + (x + y) \end{array}$$

As the extended TRS contains all original rules, we have again the previous $\mathsf{A}$-critical pair:

$$x + z \quad {}_{\{1\},\mathsf{A}}\!\leftarrow\!\rtimes\!\rightarrow_{\{2\}} \quad (x + \mathsf{0}) + z$$

Since $(x + \mathsf{0}) + z \rightarrow_{\mathsf{Ext}_{\mathsf{A}}(\{1\}),\mathsf{A}} x + z$ holds, the pair is joinable. Similarly, one can verify that all other $\mathsf{A}$-critical pairs are joinable. Therefore, $\mathsf{CR}(\mathsf{Ext}_{\mathsf{A}}(\{1\}), \mathsf{A})$ is concluded by Theorem 1. Finally, confluence of $\mathcal{R}_1$ is established.

### 3.3 Commutative Unification

Commutative unification also benefits from left-linearity. We define $\mathcal{U}^{\mathsf{C}}_{\mathcal{E}}(s \approx t)$ as $\{\mu \mid s \mathrel{-\!\!\circ\!\!\!\twoheadrightarrow}_{\mathsf{C}} s' \text{ and } \mu \in \mathcal{U}_{\mathcal{E}}(s' \approx t) \text{ for some } s'\}$.

**Lemma 9.** *Suppose* $\mathsf{C}$ *and* $\mathcal{E} \cup \mathcal{E}^{-1}$ *commute. If* $\mathcal{V}\mathsf{ar}(s) \cap \mathcal{V}\mathsf{ar}(t) = \varnothing$ *and* $s$ *is linear then* $\mathcal{U}^{\mathsf{C}}_{\mathcal{E}}(s \approx t)$ *is a complete set of* $\mathsf{C} \cup \mathcal{E}$*-unifiers for* $s$ *and* $t$*.*

*Proof.* Since it is trivial that $\mathcal{U}^{\mathsf{C}}_{\mathcal{E}}(s \approx t)$ consists of $\mathcal{E} \cup \mathsf{C}$-unifiers, we only show completeness of the set. Let $s\sigma \leftrightarrow^*_{\mathsf{C}\cup\mathcal{E}} t\sigma$. One can show $(\leftrightarrow_{\mathcal{E}} \cup \rightarrow_{\mathsf{C}})^* \subseteq {}_{\mathsf{C}}^{*}\!\!\leftarrow \cdot \leftrightarrow^*_{\mathcal{E}}$ by using induction, commutation, and the reversibility of $\mathsf{C}$. Thus, $s\sigma \; {}_{\mathsf{C}}^{*}\!\!\leftarrow u \leftrightarrow^*_{\mathcal{E}} t\sigma$ for some $u$. Since $\mathrel{-\!\!\circ\!\!\!\twoheadrightarrow}_{\mathsf{C}} = \leftrightarrow^*_{\mathsf{C}}$ holds, $s\sigma \mathrel{-\!\!\circ\!\!\!\twoheadrightarrow}_{\mathsf{C}} u$. Due to the linearity of $s$ there are $s'$ and $\sigma'$ such that $u = s'\sigma'$, $s \mathrel{-\!\!\circ\!\!\!\twoheadrightarrow}_{\mathsf{C}} s'$, and $x\sigma \mathrel{-\!\!\circ\!\!\!\twoheadrightarrow}_{\mathsf{C}} x\sigma'$ for all variables $x$. We define the substitution $\mu$ as follows:

$$\mu = \{x \mapsto x\sigma \mid x \in \mathcal{V}\mathsf{ar}(s)\} \cup \{x \mapsto x\sigma' \mid x \in \mathcal{V}\mathsf{ar}(t)\}$$

Since $s$ and $t$ share no variables, $\mu$ is well-defined. By the definition we obtain $s'\mu \leftrightarrow^*_{\mathcal{E}} t\mu$. □

Since $\mathsf{A}$ and $\mathsf{C}$ are left-linear TRSs that share no function symbols, their commutation can be proved (by using e.g. Theorem 11 in the next section). So Lemma 9 gives a way to compute $\mathsf{A} \cup \mathsf{C}$-critical pairs.

*Example 10.* Consider the left-linear TRS $\mathcal{R}_2$ with $\mathcal{F}_{\mathsf{A}} = \{*\}$ and $\mathcal{F}_{\mathsf{C}} = \{\mathsf{eq}\}$:

$$\begin{array}{lll}
1\colon \mathsf{eq}(\mathsf{a},\mathsf{a}) \rightarrow \mathsf{T} & 3\colon \mathsf{eq}(\mathsf{a} * x, y * \mathsf{a}) \rightarrow \mathsf{eq}(x,y) & 5\colon (x * y) * z \rightarrow x * (y * z) \\
2\colon \mathsf{eq}(\mathsf{a}, x * y) \rightarrow \mathsf{F} & 4\colon \mathsf{eq}(x,y) \rightarrow \mathsf{eq}(y,x) & 6\colon x * (y * z) \rightarrow (x * y) * z
\end{array}$$

Let $\mathcal{R} = \{1,2,3\}$ and $\mathcal{E} = \{4,5,6\}$. Note that $\mathcal{E} = \mathsf{C} \cup \mathsf{A}$. It is sufficient to show $\mathsf{CR}(\mathsf{Ext}_{\mathsf{A}}(\mathcal{R}), \mathcal{E})$. We can use AC-RPO to prove termination of $\mathcal{R}/\mathcal{E}$, which is equivalent to that of $\mathsf{Ext}_{\mathsf{A}}(\mathcal{R})/\mathcal{E}$ due to the identity in Lemma 6. Let $s = \mathsf{eq}(\mathsf{a} * x, y * \mathsf{a})$ and $t = \mathsf{eq}(\mathsf{a} * x', y' * \mathsf{a})$. A complete set of their $\mathsf{A} \cup \mathsf{C}$-unifiers is:

$$\begin{aligned}
\mathcal{U}^{\mathsf{C}}_{\mathsf{A}}(s \approx t) &= \langle s \approx t \rangle \cup \langle \mathsf{eq}(y * \mathsf{a}, \mathsf{a} * x) \approx \mathsf{eq}(\mathsf{a} * x', y' * \mathsf{a}) \rangle \\
&= \left\{ \begin{array}{llll}
\{x \mapsto x', & y \mapsto y'\}, & & \\
\{x \mapsto \mathsf{a}, & y \mapsto \mathsf{a}, & x' \mapsto \mathsf{a}, & y' \mapsto \mathsf{a}\} \\
\{x \mapsto y' * \mathsf{a}, & y \mapsto \mathsf{a} * y, & x' \mapsto y * \mathsf{a}, & y' \mapsto \mathsf{a} * y'\}
\end{array} \right\}
\end{aligned}$$

In this way we can compute complete sets to induce the set of all $\mathcal{E}$-critical pairs. Since all pairs are joinable, $\mathsf{CR}(\mathcal{R}, \mathcal{E})$ is concluded.

## 4 Commutation

### 4.1 Commutation Criteria

Our tool employs three commutation criteria. The first commutation criterion is the *development closedness theorem* [9–12].

**Theorem 11 (Development Closedness).** *Left-linear TRSs $\mathcal{R}$ and $\mathcal{S}$ commute if the inclusions ${}_{\mathcal{R}}\!\leftarrow\rtimes\rightarrow_{\mathcal{S}} \subseteq \mathrel{-\!\!\!\circ\!\!\!\rightarrow}_{\mathcal{S}}$ and ${}_{\mathcal{R}}\!\leftarrow\ltimes\rightarrow_{\mathcal{S}} \subseteq \rightarrow_{\mathcal{S}}^{*} \cdot \mathrel{\leftarrow\!\!\!\circ\!\!\!-}_{\mathcal{R}}$ hold.* □

The second criterion is the commutation version of the confluence criterion based on rule labeling with weight function [13,14].

**Definition 12.** *A* weight function *$w$ is a function from $\mathcal{F}$ to $\mathbb{N}$. The weight $w(C)$ of a context $C$ is defined as follows:*

$$w(C) = \begin{cases} 0 & \text{if } C = \Box \\ w(f) + w(C') & \text{if } C = f(t_1, \ldots, C', \ldots, t_n) \text{ with a context } C' \end{cases}$$

*The weight is* admissible *for a TRS $\mathcal{R}$ if*

$$\{w(C) \mid C[x] = \ell\} \geqslant^{\mathrm{mul}} \{w(C) \mid C[x] = r\}$$

*holds for all $\ell \to r \in \mathcal{R}$ and $x \in \mathcal{V}\mathsf{ar}(r)$. Here $\geqslant^{\mathrm{mul}}$ stands for the multiset extension of the standard order $>$ on $\mathbb{N}$ (see e.g. [2]). A* rule labeling *$\phi$ for a TRS $\mathcal{R}$ is a function from $\mathcal{R}$ to $\mathbb{N}$. The labeled step $\xrightarrow{\alpha}_{\mathcal{R}}$ is defined as follows: $s \rightarrow_{\mathcal{R},(k,m)} t$ if there are a rule $\ell \to r$, a context $C$, and a substitution $\sigma$ such that $s = C[\ell\sigma]$, $t = C[r\sigma]$, and $\alpha = (w(C), \phi(\ell \to r))$.*

In the next theorem we use the following abbreviations for labeled steps:

$$\xrightarrow{I} = \bigcup_{\alpha \in I} \xrightarrow{\alpha} \qquad \curlyvee\alpha = \{\beta \in I \mid \alpha \succ \beta\} \qquad \curlyvee\alpha\beta = (\curlyvee\alpha) \cup (\curlyvee\beta)$$

where, $\succ$ stands for the lexicographic order on $\mathbb{N} \times \mathbb{N}$.

**Theorem 13 (Rule Labeling).** *Left-linear TRSs $\mathcal{R}$ and $\mathcal{S}$ commute if there are an admissible weight function $w$ and a rule labeling $\phi$ for $\mathcal{R} \cup \mathcal{S}$ such that*

$$({}_{\mathcal{R}}\!\xleftarrow{\alpha}\rtimes\xrightarrow{\beta}_{\mathcal{S}}) \cup ({}_{\mathcal{R}}\!\xleftarrow{\alpha}\ltimes\xrightarrow{\beta}_{\mathcal{S}}) \subseteq \xrightarrow{\curlyvee\alpha}{}_{\mathcal{S}}^{*} \cdot \xrightarrow{\beta}{}_{\mathcal{S}}^{=} \cdot \xrightarrow{\curlyvee\alpha\beta}{}_{\mathcal{S}}^{*} \cdot {}_{\mathcal{R}}^{*}\!\xleftarrow{\curlyvee\alpha\beta} \cdot {}_{\mathcal{R}}^{=}\!\xleftarrow{\alpha} \cdot {}_{\mathcal{R}}^{*}\!\xleftarrow{\curlyvee\beta}$$

*for all pairs $\alpha, \beta \in \mathbb{N} \times \mathbb{N}$.* □

The final criterion is a trivial adaptation of Theorem 1 to the commutation property, integrating Lemmata 6, 8, and 9.

**Theorem 14 (Church-Rosser Modulo).** *Let $\mathcal{R}, \mathcal{S}$ be left-linear TRSs and $\mathcal{E} \in \{\mathsf{A}, \mathsf{AC}\}$ such that $\mathcal{R}/\mathcal{E}'$ is terminating for $\mathcal{E}' = \mathcal{E} \cup \mathsf{C}$. The TRSs $\mathcal{R} \cup \mathcal{E}'$ and $\mathcal{S} \cup \mathcal{E}'$ commute if and only if the inclusion holds:*

$$({}_{\mathcal{R}',\mathcal{E}'}\!\leftarrow\rtimes\rightarrow_{\mathcal{S}' \cup \mathcal{E}'}) \cup ({}_{\mathcal{R}' \cup \mathcal{E}'}\!\leftarrow\ltimes\rightarrow_{\mathcal{S}',\mathcal{E}'}) \subseteq \rightarrow_{\mathcal{S}',\mathcal{E}'}^{*} \cdot \leftrightarrow_{\mathcal{E}'}^{*} \cdot {}_{\mathcal{R}',\mathcal{E}'}^{*}\!\leftarrow$$

*Here $\mathcal{R}' = \mathsf{Ext}_{\mathcal{E}}(\mathcal{R})$ and $\mathcal{S}' = \mathsf{Ext}_{\mathcal{E}}(\mathcal{S})$.* □

Note that our tool uses the algorithm in [15] for AC unification and flattened term representation for overcoming the coherence problem of AC-rewriting. Since we use the dedicated algorithms for A and AC unification, currently we cannot employ Theorem 1 with $\mathcal{E} = \mathsf{A} \cup \mathsf{AC}$.

### 4.2 Commutation Theorem

The next theorem is known as Hindley's Commutation Theorem [1].

**Theorem 15 (Commutation Theorem).** *If $\xrightarrow{\alpha}$ and $\xrightarrow{\beta}$ commute for all $\alpha \in I$ and $\beta \in J$ then $\xrightarrow{I}$ and $\xrightarrow{J}$ commute.* □

*Example 16.* Consider the left-linear TRS $\mathcal{R}_3$:

1: $\mathsf{0} \times y \to \mathsf{nil}$

2: $\mathsf{s}(x) \times y \to y \mathbin{+\!\!+} (x \times y)$

3: $\mathsf{hd}(\mathsf{c}(x)) \to x$

4: $\mathsf{hd}(\mathsf{c}(x) \mathbin{+\!\!+} y) \to x$

5: $\mathsf{nil} \mathbin{+\!\!+} x \to x$

6: $x \mathbin{+\!\!+} \mathsf{nil} \to x$

7: $x \mathbin{+\!\!+} (y \mathbin{+\!\!+} z) \to (x \mathbin{+\!\!+} y) \mathbin{+\!\!+} z$

8: $(x \mathbin{+\!\!+} y) \mathbin{+\!\!+} z \to x \mathbin{+\!\!+} (y \mathbin{+\!\!+} z)$

9: $\mathsf{from}(x) \to x : \mathsf{from}(\mathsf{s}(x))$

By using the Commutation Theorem we show self-commutation of $\mathcal{R}_3$:

(i) Self-commutation of $\{1, \ldots, 8\}$ follows from Theorem 14.
(ii) Commutation of $\{1, \ldots, 8\}$ and $\{9\}$ follows from Theorem 11.
(iii) Self-commutation of $\{9\}$ is proved by Theorem 11.

Hence, $\mathcal{R}_3$ is confluent.

It is a non-trivial task to find suitable commuting subsystems from an exponential number of candidates. In order to address the problem we introduce a decomposition method based on *composability*, which was introduced by Ohlebusch [16]. Let $\mathcal{R}$ be a TRS. We write $\mathcal{F}_\mathcal{R}$, $\mathcal{D}_\mathcal{R}$, and $\mathcal{C}_\mathcal{R}$ for the following sets:

$$\mathcal{F}_\mathcal{R} = \bigcup_{\ell \to r \in \mathcal{R}} \mathcal{F}\mathsf{un}(\ell) \cup \mathcal{F}\mathsf{un}(r) \qquad \mathcal{D}_\mathcal{R} = \{\mathrm{root}(\ell) \mid \ell \to r \in \mathcal{R}\} \qquad \mathcal{C}_\mathcal{R} = \mathcal{F}_\mathcal{R} \setminus \mathcal{D}_\mathcal{R}$$

**Definition 17.** *We say that TRSs $\mathcal{R}$ and $\mathcal{S}$ are* composable *if $\mathcal{C}_\mathcal{R} \cap \mathcal{D}_\mathcal{S} = \mathcal{C}_\mathcal{S} \cap \mathcal{D}_\mathcal{R} = \varnothing$ and $\{\ell \to r \in \mathcal{R} \cup \mathcal{S} \mid \mathrm{root}(\ell) \in \mathcal{D}_\mathcal{R} \cup \mathcal{D}_\mathcal{S}\} \subseteq \mathcal{R} \cap \mathcal{S}$.*

Ohlebusch [16] posed the following question.

*Question 18.* Are left-linear composable TRSs $\mathcal{R}$ and $\mathcal{S}$ confluent if and only if $\mathcal{R} \cup \mathcal{S}$ is confluent?

Although the question still remains open, the following variation is valid.

**Theorem 19.** *Commuting composable TRSs $\mathcal{R}$ and $\mathcal{S}$ are confluent if and only if $\mathcal{R} \cup \mathcal{S}$ is confluent.* □

*Example 20.* Recall the TRS $\mathcal{R}_3$ from Example 16. The TRS is the union of the three commuting composable subsystems: $\{1, 2, 5, 6, 7, 8\}$, $\{3, 4, 5, 6, 7, 8\}$, and $\{9\}$. Confluence of each subsystem can be proved in the same method used in the previous example. Hence, $\mathcal{R}_3$ is confluent.

## 5 Implementation

The confluence tool CoLL consists of about 5,000 lines of OCaml code. Given an input TRS, the tool first performs the next trivial *redundant rule elimination.*

**Theorem 21.** *Let $\mathcal{R}$ and $\mathcal{S}$ be TRSs with $\mathcal{S} \subseteq \to_{\mathcal{R}}^*$. The TRS $\mathcal{R} \cup \mathcal{S}$ is confluent if and only if $\mathcal{R}$ is confluent.* □

*Example 22.* We illustrate the elimination technique with a small example taken from the Confluence Problem Database (Cops)[1]. Consider the TRS:

$$1\colon \mathsf{f}(x) \to \mathsf{g}(x, \mathsf{f}(x)) \qquad 2\colon \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(x)))) \to \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{g}(x, \mathsf{f}(x)))))$$

Since $\{2\} \subseteq \to_{\{1\}}^*$ holds, we eliminate the redundant rule 2. Confluence of the simplified system $\{1\}$ is easily shown by Theorem 11. Note that CoLL cannot prove confluence without using the elimination technique.

Next, the tool employs Theorem 19 to split the simplified TRS into commuting composable subsystems $\mathcal{R}_1, \ldots, \mathcal{R}_n$. For each subsystem $\mathcal{R}_i$ the tool performs the non-confluence test of [17, Lemma 1]. If non-confluence is detected, the tool outputs NO (non-confluence is proved). Otherwise, the tool uses the Commutation Theorem together with the three commutation criteria (Theorems 11, 13, and 14) to determine self-commutation of $\mathcal{R}_i$. Suitable commuting subsystems are searched by enumeration. It outputs YES (confluence is proved) if all of $\mathcal{R}_1, \ldots, \mathcal{R}_n$ are confluent. Concerning automation, we employed AC-RPO for checking termination of $\mathcal{R}/\mathcal{E}$ automatically. Automation of Theorem 13 is based on the SAT encoding technique of [18].

We tested the presented techniques on 188 left-linear TRSs in Cops Nos. 1–425, where we ruled out duplicated problems.[2] The tests were run on a PC equipped with an Intel Core i7-4500U CPU with 1.8 GHz and 3.8 GB of RAM using a timeout of 120 sec. For the sake of comparison we also ran the tools that participated in the 3rd Confluence Competition: ACP v0.5 [9], CSI v0.4.1 [17], and Saigawa v1.7[3]. The first table in Fig. 1 summarizes the results. The first three indicate the results of each commutation criterion without using the Commutation Theorem. The second table indicates the results of individual theories for Theorem 14. The row ‘all three’ in the first table is the summation of their results, and ‘all with elimination’ is the same but the elimination technique is enabled. The row CoLL corresponds to the strategy stated above. On our problem set, all confluence proofs by Saigawa are covered by CoLL. The results of CoLL, ACP, and CSI are incomparable.

[1] http://cops.uibk.ac.at/.

[2] All problems and results are available at the tool website (see the URL in Section 1).

[3] http://www.jaist.ac.jp/project/saigawa/.

| | YES | NO | timeout |
|---|---|---|---|
| Church-Rosser modulo | **93** | **8** | 1 |
| development closed | **17** | **0** | 0 |
| rule labeling | **58** | **0** | 26 |
| all three | **125** | **8** | – |
| all with elimination | **136** | **9** | – |
| CoLL | **137** | **16** | 21 |
| ACP | **134** | **41** | 0 |
| CSI | **118** | **38** | 11 |
| Saigawa | **105** | **16** | 17 |

| $\mathcal{E}$ | YES | NO | timeout |
|---|---|---|---|
| $\varnothing$ | **18** | **8** | 0 |
| A | **24** | **0** | 0 |
| C | **42** | **8** | 0 |
| AC | **64** | **8** | 1 |
| C $\uplus$ AC | **88** | **8** | 1 |
| A $\uplus$ C $\uplus$ AC | **93** | **8** | 1 |

**Fig. 1.** Experimental results.

## 6 Conclusion

We presented the new confluence tool CoLL for left-linear TRSs, which proves confluence via commutation. Our primary contribution is automation of Jouannaud and Kirchner's Church-Rosser modulo criterion for associativity and/or commutativity theory, where left-linearity is exploited in several ways.

We briefly compare CoLL with existing confluence tools. CRC 3 [19] is a powerful Church-Rosser checker for Maude and supports the Church-Rosser modulo theorem for any combination of associativity, commutativity, and/or identity theories, except associativity theory. When handling TRSs that contain reversible rules, ACP [9] employs *reduction-preserving completion* [20]. This method effectively works for C and AC rules, but not for associativity rules. ACP and CSI [17] employ layer-preserving decomposition [16] to split a TRS into subsystems. The technique is incomparable to Theorem 19. If Question 18 is affirmatively solved, it generalizes the two techniques for the class of left-linear TRSs. Finally, CoLL was designed for a complement of Saigawa. The two tools will be merged in the next version.

As future work we plan to investigate whether Theorem 19 can be generalized to cover a subclass of *hierarchical combination* [16]. Another interesting direction is the modularity of the commutation property. Since confluence is a modular property [21], it is closed under *signature extension*. Contrary to our expectation, (even local) commutation is *not* signature extensible. Consider the TRSs $\mathcal{R}$ and $\mathcal{S}$ over the signature $\mathcal{F} = \{\mathsf{f}^{(2)}, \mathsf{a}^{(0)}, \mathsf{b}^{(0)}\}$:

$$\mathcal{R} = \{\, \mathsf{a} \to \mathsf{b} \,\} \qquad \mathcal{S} = \left\{ \begin{array}{rrr} \mathsf{f}(x,x) \to \mathsf{b}, & \mathsf{f}(\mathsf{a},x) \to \mathsf{b}, & \mathsf{f}(x,\mathsf{a}) \to \mathsf{b} \\ & \mathsf{f}(\mathsf{b},x) \to \mathsf{b}, & \mathsf{f}(x,\mathsf{b}) \to \mathsf{b} \end{array} \right\}$$

Since $C[t] \to_{\mathcal{S}}^{*} \mathsf{b}$ holds for all contexts $C$ and $t \in \{\mathsf{a}, \mathsf{b}\}$, we obtain the strong commutation ${}_{\mathcal{R}}\!\leftarrow \cdot \to_{\mathcal{S}} \subseteq \to_{\mathcal{S}}^{*} \cdot {}_{\mathcal{R}}^{=}\!\leftarrow$, which entails commutation of $\mathcal{R}$ and $\mathcal{S}$. However, if one extends the signature to $\mathcal{F} \cup \{\mathsf{g}^{(1)}\}$, the local peak $\mathsf{f}(\mathsf{g}(\mathsf{b}), \mathsf{g}(\mathsf{a}))\ {}_{\mathcal{R}}\!\leftarrow \mathsf{f}(\mathsf{g}(\mathsf{a}), \mathsf{g}(\mathsf{a})) \to_{\mathcal{S}} \mathsf{b}$ no longer commutes. We conjecture that (local) commutation is closed under signature extension for left-linear TRSs.

**Acknowledgements.** We are grateful for the detailed comments of the anonymous reviewers, which helped us to improve the presentation.

## References

1. Hindley, J.R.: The Church-Rosser Property and a Result in Combinatory Logic. Ph.D. thesis, University of Newcastle-upon-Tyne (1964)
2. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press, Cambridge (1998)
3. Jouannaud, J.P., Kirchner, H.: Completion of a set of rules modulo a set of equations. SIAM J. Comput. **15**(4), 1155–1194 (1986)
4. Rubio, A.: A fully syntactic AC-RPO. Inf. Comput. **178**(2), 515–533 (2002)
5. Plotkin, G.: Building in equational theories. Mach. Intell. **7**, 73–90 (1972)
6. Schulz, K.: Word unification and transformation of generalized equations. In: Abdulrab, H., Pecuchet, J.-P. (eds.) IWWERT 1991. LNCS, vol. 677, pp. 150–176. Springer, Heidelberg (1993)
7. Schmidt, R.A.: $E$-Unification for subsystems of $S4$. In: Nipkow, T. (ed.) RTA 1998. LNCS, vol. 1379, pp. 106–120. Springer, Heidelberg (1998)
8. Peterson, G., Stickel, M.: Complete sets of reductions for some equational theories. J. ACM **28**(2), 233–264 (1981)
9. Aoto, T., Yoshida, J., Toyama, Y.: Proving confluence of term rewriting systems automatically. In: Treinen, R. (ed.) RTA 2009. LNCS, vol. 5595, pp. 93–102. Springer, Heidelberg (2009)
10. Huet, G.: Confluent reductions: abstract properties and applications to term rewriting systems. J. ACM **27**(4), 797–821 (1980)
11. Toyama, Y.: Commutativity of term rewriting systems. In: Fuchi, K., Kott, L. (eds.) Programming of Future Generation Computers II, pp. 393–407. North-Holland, Amsterdam (1988)
12. van Oostrom, V.: Developing developments. Theoret. Comput. Sci. **175**(1), 159–181 (1997)
13. van Oostrom, V.: Confluence by decreasing diagrams. In: Voronkov, A. (ed.) RTA 2008. LNCS, vol. 5117, pp. 306–320. Springer, Heidelberg (2008)
14. Aoto, T.: Automated confluence proof by decreasing diagrams based on rule-labelling. In: Lynch, C. (ed.) RTA 2010, LIPIcs, vol. 6, pp. 7–16 (2010)
15. Pottier, L.: Minimal solutions of linear diophantine systems: bounds and algorithms. In: Book, R.V. (ed.) Rewriting Techniques and Applications. LNCS, vol. 488, pp. 162–173. Springer, Heidelberg (1991)
16. Ohlebusch, E.: Modular Properties of Composable Term Rewriting Systems. Ph.D. thesis, Universität Bielefeld (1994)
17. Zankl, H., Felgenhauer, B., Middeldorp, A.: CSI – a confluence tool. In: Bjørner, N., Sofronie-Stokkermans, V. (eds.) CADE-23. LNCS, vol. 6803, pp. 499–505. Springer, Heidelberg (2011)
18. Hirokawa, N., Middeldorp, A.: Decreasing diagrams and relative termination. J. Autom. Reason. **47**(4), 481–501 (2011)
19. Durán, F., Meseguer, J.: A Church-Rosser checker tool for conditional order-sorted equational maude specifications. In: Ölveczky, P.C. (ed.) WRLA 2010. LNCS, vol. 6381, pp. 69–85. Springer, Heidelberg (2010)
20. Aoto, T., Toyama, Y.: A reduction-preserving completion for proving confluence of non-terminating term rewriting systems. LMCS **8**(1), 1–29 (2012)
21. Toyama, Y.: On the Church-Rosser property for the direct sum of term rewriting systems. J. ACM **34**(1), 128–143 (1987)