Confluence by Strong Commutation with Disjoint Parallel Reduction

Kentaro Kikuchi
RIEC, Tohoku University, Sendai, Japan
kentaro@nue.riec.tohoku.ac.jp

Confluence is a fundamental property of rewriting systems that guarantees uniqueness of results of computation. In this paper we propose a general method for proving confluence through sufficient conditions for strong commutation with disjoint parallel reduction. The method works not only for first-order term rewriting systems but also for systems with bound variables and conditional rules.

1 Introduction

Confluence is a fundamental property of rewriting systems and is important in their applications. For first-order term rewriting systems (TRSs), confluence has been well studied, and many criteria to ensure confluence have been developed. For rewriting systems with bound variables, confluence has also been well studied but in most cases only for individual systems. The aim of this work is to provide uniform methods for proving confluence for classes of TRSs and rewriting systems with bound variables, and to extract criteria for confluence from such proofs. To this end, we also propose a format of specification of rewriting systems and notation for it in a rather different way from the standard format and notation of TRSs and traditional higher-order rewriting frameworks.

Our starting point is what is called an inductive confluence (or commutation) proof for (mutually) orthogonal TRSs as explained in [1, pp. 208–211]. This method uses an inductive definition of disjoint parallel reduction which reduces redexes at disjoint positions simultaneously. In [1], it is shown that such defined disjoint parallel reduction satisfies the (commuting) diamond property. Our approach is different in that we prove, instead of the diamond property, strong commutation of usual one-step reduction with disjoint parallel reduction. More precisely, we extract sufficient conditions for strong commutation from the proof by induction on the derivation of disjoint parallel reduction. The sufficient conditions can be applied to all rewriting systems in our format including non-orthogonal systems and even systems with bound variables and conditional rules.

Our format of specification of rewriting systems is obtained by generalising the conventional way of specifying reduction rules in the field of λ-calculus. It has the following features:

(a) Reduction rules are described as infinite sets using meta-variables for terms. This is common in the field of λ-calculus while in the field of TRSs rewrite rules are defined as pairs of first-order terms with object-variables. For finite representations of reduction rules, we introduce the notion of *reduction rule schema* which employs indexed holes rather than names for meta-variables.

(b) The notion of *grammatical context* is used to define disjoint parallel reduction in a general way. This notion has been used in a previous paper [16].

(c) The way to evaluate the conditional part of a reduction rule is assumed to be known.

(d) In the present paper, it is assumed that the RHS of a rule does not include meta-operation such as capture-free substitution.

Submitted to:
WPTE 2017

Closely related formalisms for rewriting systems in the literature are conditional λ-calculus [17] and (context-sensitive) conditional expression reduction systems (CCERSs) [6, 11]. In conditional λ-calculus the notion of skeleton corresponds to our notion of reduction rule schema, but it uses meta-variables and capturing substitution instead of holes and hole-filling. We find the use of the latter helpful in writing proofs succinctly and in paying attention to capture of free variables in terms with which holes are filled. Also, the conditional part of a rule is not included in a skeleton, while it is included in a reduction rule schema as a syntactic expression. In both formalisms of conditional λ-calculus and CCERSs, confluence has been studied only for (slight extensions of) orthogonal systems.

Strong commutation has often been used to prove confluence for individual rewriting systems with bound variables (e.g. [14]) or classes of rewriting systems where β-rule is the only reduction rule with bound variables (e.g. [3]). For first-order TRSs, an automated confluence tool based on commutation criteria has been presented in [15].

The paper is organised as follows. In Section 2 we introduce our specification format of rewriting systems, and define disjoint parallel reduction in a general way. In Section 3 we present sufficient conditions for strong commutation to prove confluence. In Section 4 we discuss rewriting systems with bound variables and conditional rules. In Section 5 we conclude with referring to related work.

2 Disjoint parallel reduction

We start by introducing notations for specifying rewriting systems to present disjoint parallel reduction in a general way as a binary relation on terms that are inductively defined. Here we treat rewriting systems without conditional rules. Rewriting systems with conditional rules will be discussed in Section 4.

First we define the notions of grammatical context (Definition 2.2) and reduction rule schema (Definition 2.5), using examples from combinatory logic (CL).

Definition 2.1 (Grammar of CL)**.** *The set of terms of combinatory logic (CL) is defined by the following grammar:*

$$M,N ::= x \mid S \mid K \mid M \cdot N$$

where x ranges over a denumerable set of variables.

Definition 2.2 (Grammatical context)**.** *Given the grammar of terms of a system $\mathscr{L}$, the* grammatical context *of $\mathscr{L}$, denoted $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[\,]$, is defined by replacing, in (the RHS of) the grammar, each meta-variable by a hole.*

Example 2.3. For CL, $\mathscr{C}^{\mathsf{grm}}_{CL}[\,]$ is defined by

$$\mathscr{C}^{\mathsf{grm}}_{CL}[\,] ::= x \mid S \mid K \mid \Box_1 \cdot \Box_2$$

As in usual contexts, we use notations like $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_1, M_2]$ to denote the term obtained from a grammatical context $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[\,]$ with two holes by filling them with M_1 and M_2. We also use $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_i]_i$ to denote the term obtained from $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[\,]$ with an arbitrary number of holes by filling them with M_i's.

Definition 2.4 (Reduction system of CL)**.** *The reduction rules of CL are:*

$$\begin{array}{lrcl} (S) & ((S \cdot M) \cdot N) \cdot P & \to & (M \cdot P) \cdot (N \cdot P) \\ (K) & (K \cdot M) \cdot N & \to & M \end{array}$$

The reduction relation $\longrightarrow_S$ is defined by the contextual closure of the rule (S). We use $\longrightarrow_S^=$ for its reflexive closure, and $\longrightarrow_S^*$ for its reflexive transitive closure. These kinds of notations are also used for the notions of other reductions in this paper.

Definition 2.5 (Reduction rule schema)**.** *For a reduction rule* (ρ) *of a system* $\mathscr{L}$*, its* reduction rule schema *is defined as the pair* $\langle \mathscr{C}^{\mathsf{left}}_{\rho}[\,], \mathscr{C}^{\mathsf{right}}_{\rho}[\,]\rangle$ *where* $\mathscr{C}^{\mathsf{left}}_{\rho}[\,]$ *is obtained by replacing, in the LHS of the rule* (ρ)*, each meta-variable by a hole, and* $\mathscr{C}^{\mathsf{right}}_{\rho}[\,]$ *by replacing, in the RHS of* (ρ)*, each meta-variable by the respective hole.*

We treat non-left-linear rules as conditional rules by the technique well-known as, e.g. de Vrijer's conditional linearisation.

Example 2.6. For the rule (S) of CL,

$$\mathscr{C}^{\mathsf{left}}_{S}[\,] \equiv ((S\cdot\Box_1)\cdot\Box_2)\cdot\Box_3 \quad \text{and} \quad \mathscr{C}^{\mathsf{right}}_{S}[\,] \equiv (\Box_1\cdot\Box_3)\cdot(\Box_2\cdot\Box_3)$$

For the rule (K) of CL,

$$\mathscr{C}^{\mathsf{left}}_{K}[\,] \equiv (K\cdot\Box_1)\cdot\Box_2 \quad \text{and} \quad \mathscr{C}^{\mathsf{right}}_{K}[\,] \equiv \Box_1$$

As in the case of grammatical contexts, we use notations such as $\mathscr{C}^{\mathsf{left}}_{\rho}[M_1,M_2]$, $\mathscr{C}^{\mathsf{left}}_{\rho}[M_i]_i$ ($\mathscr{C}^{\mathsf{right}}_{\rho}[M_1, M_2]$, $\mathscr{C}^{\mathsf{right}}_{\rho}[M_i]_i$) to denote the terms obtained from $\mathscr{C}^{\mathsf{left}}_{\rho}[\,]$ ($\mathscr{C}^{\mathsf{right}}_{\rho}[\,]$, respectively) by filling the holes with M_i's.

Example 2.7. For the rule (S) of CL,

$$\mathscr{C}^{\mathsf{left}}_{S}[K,x,K] \equiv ((S\cdot K)\cdot x)\cdot K \quad \text{and} \quad \mathscr{C}^{\mathsf{right}}_{S}[K,x,K] \equiv (K\cdot K)\cdot(x\cdot K)$$

For the rule (K) of CL,

$$\mathscr{C}^{\mathsf{left}}_{K}[S,x\cdot K] \equiv (K\cdot S)\cdot(x\cdot K) \quad \text{and} \quad \mathscr{C}^{\mathsf{right}}_{K}[S,x\cdot K] \equiv S$$

Note that $\mathscr{C}^{\mathsf{right}}_{\rho}[M_i]_i$ has the same number of arguments (i.e. M_i's in $[M_i]_i$) as $\mathscr{C}^{\mathsf{left}}_{\rho}[M_i]_i$ has, even if some of the M_i's are copied or do not occur in $\mathscr{C}^{\mathsf{right}}_{\rho}[M_i]_i$.

Now we define the notion of disjoint parallel reduction. The same (but differently difined) notion for first-order TRSs is called parallel-disjoint reduction in [9].

Definition 2.8 (Disjoint parallel reduction)**.** *Let* (ρ) *be a reduction rule of a system* $\mathscr{L}$ *with the reduction rule schema* $\langle \mathscr{C}^{\mathsf{left}}_{\rho}[\,], \mathscr{C}^{\mathsf{right}}_{\rho}[\,]\rangle$*. Then the disjoint parallel* ρ*-reduction (dpr* ρ*-reduction for short)* $\multimap\!\!\!\rightarrow_{\rho}$ *is the relation inductively defined by the following inference rules:*

$$\frac{[M_i \multimap\!\!\!\rightarrow_{\rho} N_i]_i}{\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_i]_i \multimap\!\!\!\rightarrow_{\rho} \mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[N_i]_i}\ (\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}}) \qquad \frac{}{\mathscr{C}^{\mathsf{left}}_{\rho}[M_i]_i \multimap\!\!\!\rightarrow_{\rho} \mathscr{C}^{\mathsf{right}}_{\rho}[M_i]_i}\ (\mathsf{dpr}_{\rho})$$

where $[M_i \multimap\!\!\!\rightarrow_{\rho} N_i]_i$ *denotes the premisses* $M_1 \multimap\!\!\!\rightarrow_{\rho} N_1,\ \ldots,\ M_n \multimap\!\!\!\rightarrow_{\rho} N_n$*, the number of which depends on the form of* $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[\,]$*.*

Lemma 2.9. *For any dpr* ρ*-reduction* $\multimap\!\!\!\rightarrow_{\rho}$ *in a system* $\mathscr{L}$ *and any term* M *of* $\mathscr{L}$*,* $M \multimap\!\!\!\rightarrow_{\rho} M$*.*

Proof. $M \multimap\!\!\!\rightarrow_{\rho} M$ is derived using the rule $(\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$ only. Formally, the proof is by induction on the structure of M. □

Lemma 2.10. *For any dpr* ρ*-reduction* $\multimap\!\!\!\rightarrow_{\rho}$*, the following hold.*

1. *If* $M \longrightarrow_{\rho} N$ *then* $M \multimap\!\!\!\rightarrow_{\rho} N$*.*
2. *If* $M \multimap\!\!\!\rightarrow_{\rho} N$ *then* $M \longrightarrow^{*}_{\rho} N$*.*

Proof.

1. If $M \longrightarrow_{\rho} N$ then $M \multimap\!\!\!\rightarrow_{\rho} N$ is derived using the rule (dpr_{ρ}) exactly once. Formally, the proof is by induction on the structure of M.
2. By induction on the derivation of $M \multimap\!\!\!\rightarrow_{\rho} N$. □

3 Commutation via disjoint parallel reduction

Our basic strategy for proving confluence is to show commutation of each two rules of the system. We first recall definitions and lemmas on commutation (cf. [1, pp. 31–33]).

Definition 3.1. *Let $\longrightarrow_{R_1}$ and $\longrightarrow_{R_2}$ be two binary relations on terms of a system $\mathscr{L}$.*

1. $\longrightarrow_{R_1}$ *and* $\longrightarrow_{R_2}$ commute $\stackrel{\text{def}}{\Longleftrightarrow}$ *if $M \longrightarrow^*_{R_1} M_1$ and $M \longrightarrow^*_{R_2} M_2$ then there exists N such that $M_1 \longrightarrow^*_{R_2} N$ and $M_2 \longrightarrow^*_{R_1} N$.*
2. $\longrightarrow_{R_1}$ *is* confluent $\stackrel{\text{def}}{\Longleftrightarrow}$ $\longrightarrow_{R_1}$ *and* $\longrightarrow_{R_1}$ *commute.*
3. $\longrightarrow_{R_1}$ strongly commutes with $\longrightarrow_{R_2}$ $\stackrel{\text{def}}{\Longleftrightarrow}$ *if $M \longrightarrow_{R_1} M_1$ and $M \longrightarrow_{R_2} M_2$ then there exists N such that $M_1 \longrightarrow^{=}_{R_2} N$ and $M_2 \longrightarrow^*_{R_1} N$.*

Lemma 3.2 ([8])**.**

1. *If $\longrightarrow_{R_1}$ and $\longrightarrow_{R_2}$ are confluent and commute then $\longrightarrow_{R_1} \cup \longrightarrow_{R_2}$ is also confluent.*
2. *If $\longrightarrow_{R_i}$ and $\longrightarrow_{R_j}$ commute for any $i, j \in I$ then $\bigcup_{i \in I} \longrightarrow_{R_i}$ is confluent.*

Lemma 3.3 ([8])**.** *If $\longrightarrow_{R_1}$ strongly commutes with $\longrightarrow_{R_2}$ then they commute.*

To show that $\longrightarrow_{\rho_1}$ and $\longrightarrow_{\rho_2}$ commute in a system $\mathscr{L}$, we often use Lemma 3.3 taking $\longrightarrow_{\rho_1}$ as $\longrightarrow_{R_1}$ and the dpr ρ_2-reduction $\mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2}$ as $\longrightarrow_{R_2}$. In the following we give a sufficient condition for strong commutation of $\longrightarrow_{\rho_1}$ with $\mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2}$, which implies commutation of $\longrightarrow_{\rho_1}$ and $\longrightarrow_{\rho_2}$ since $\longrightarrow^*_{\rho_2} = \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}^*_{\rho_2}$ by Lemma 2.10.

Theorem 3.4 (Sufficient condition for strong commutation)**.** *Let (ρ_1) and (ρ_2) be reduction rules of a system $\mathscr{L}$. Suppose that $\longrightarrow_{\rho_1}$ and $\mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2}$ satisfy the following conditions:*

(sc_1) *If $M \longrightarrow_{\rho_1} M'$ with the ρ_1-redex at the root, and $M \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N$ is derived with $(\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$ as the last applied rule, then there exists N' such that $M' \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N'$ and $N \longrightarrow^*_{\rho_1} N'$.*

(sc_2) *If $\mathscr{C}^{\mathsf{left}}_{\rho_2}[M_i]_i \longrightarrow_{\rho_1} M'$ with the ρ_1-redex not in any M_i, then there exists N' such that $M' \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N'$ and $\mathscr{C}^{\mathsf{right}}_{\rho_2}[M_i]_i \longrightarrow^*_{\rho_1} N'$.*

Then $\longrightarrow_{\rho_1}$ strongly commutes with $\mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2}$.

$$\begin{array}{ccc} M & \xrightarrow[\rho_2]{\;\shortparallel\;} & N \\ \big\downarrow{\scriptstyle \rho_1} & & {\scriptstyle *}\big\downarrow{\scriptstyle \rho_1} \\ M' & \xrightarrow[\rho_2]{\;\shortparallel\;} & N' \end{array}$$

Proof. We prove by induction on the derivation of $M \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N$ that if $M \longrightarrow_{\rho_1} M'$ and $M \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N$ then there exists N' such that $M' \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N'$ and $N \longrightarrow^*_{\rho_1} N'$.

- Suppose that the last part of the derivation of $M \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N$ has the form

$$\frac{M_1 \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N_1 \quad \cdots \quad M_n \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N_n}{\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_1,\ldots,M_n] \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} \mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[N_1,\ldots,N_n]}\ (\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$$

  First we consider the case where the reduction $M \longrightarrow_{\rho_1} M'$ takes place in $\mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_1,\ldots,M_n]$ with $M_i \longrightarrow_{\rho_1} M'_i$ for some $i \in \{1,\ldots,n\}$. Then by the induction hypothesis, there exists N'_i such that $M'_i \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} N'_i$ and $N_i \longrightarrow^*_{\rho_1} N'_i$. Hence by applying the rule $(\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$, we have

$$M' \equiv \mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[M_1,\ldots,M'_i,\ldots,M_n] \mathrel{-\!\!\shortparallel\!\!\twoheadrightarrow}_{\rho_2} \mathscr{C}^{\mathsf{grm}}_{\mathscr{L}}[N_1,\ldots,N'_i,\ldots,N_n]$$

Also, from $N_i \longrightarrow_{\rho_1}^* N_i'$ we have

$$N \equiv \mathscr{C}_{\mathscr{L}}^{\mathsf{grm}}[N_1,\ldots,N_i,\ldots,N_n] \longrightarrow_{\rho_1}^* \mathscr{C}_{\mathscr{L}}^{\mathsf{grm}}[N_1,\ldots,N_i',\ldots,N_n]$$

Thus the claim follows by taking $N' \equiv \mathscr{C}_{\mathscr{L}}^{\mathsf{grm}}[N_1,\ldots,N_i',\ldots,N_n]$.

Next we consider the case where the redex of $M \longrightarrow_{\rho_1} M'$ is not in any M_i of $\mathscr{C}_{\mathscr{L}}^{\mathsf{grm}}[M_1,\ldots,M_n]$. Then we can assume that the ρ_1-redex is at the root (from the definition of the grammar of $\mathscr{L}$). Hence the claim follows from the condition (sc_1).

- Suppose that $M \mathrel{-\!\!\shortparallel\!\!\rightarrow}_{\rho_2} N$ is derived by the rule (dpr_{ρ_2}) of the form

$$\frac{}{\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\ldots,M_n] \mathrel{-\!\!\shortparallel\!\!\rightarrow}_{\rho_2} \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\ldots,M_n]}\ (\mathsf{dpr}_{\rho_2})$$

First we consider the case where the reduction $M \longrightarrow_{\rho_1} M'$ takes place in $\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\ldots,M_n]$ with $M_i \longrightarrow_{\rho_1} M_i'$ for some $i \in \{1,\ldots,n\}$. Then by the rule (dpr_{ρ_2}), we have (noting the left-linearity of (ρ_2))

$$M' \equiv \mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\ldots,M_i',\ldots,M_n] \mathrel{-\!\!\shortparallel\!\!\rightarrow}_{\rho_2} \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\ldots,M_i',\ldots,M_n]$$

Also, we have

$$N \equiv \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\ldots,M_i,\ldots,M_n] \longrightarrow_{\rho_1}^* \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\ldots,M_i',\ldots,M_n]$$

Thus the claim follows by taking $N' \equiv \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\ldots,M_i',\ldots,M_n]$.

The case where the redex of $M \longrightarrow_{\rho_1} M'$ is not in any M_i of $\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\ldots,M_n]$ follows from the condition (sc_2). □

The above proof implicitly uses the following property: one step reduction $\longrightarrow_{\rho_1}$ can be applied at any place of a subterm occurrence including a ρ_1-redex. It would be difficult to extract similar conditions to (sc_1) and (sc_2) if we tried to show the (commuting) diamond property using $\mathrel{-\!\!\shortparallel\!\!\rightarrow}_{\rho_1}$.

It is easy to check the conditions (sc_1) and (sc_2) for the rules of CL.

Example 3.5. Let (ρ_1) and (ρ_2) be the rules (S) and (K) of CL, respectively. To check the condition (sc_1), suppose $((S\cdot M_1)\cdot M_2)\cdot M_3 \longrightarrow_S (M_1\cdot M_3)\cdot(M_2\cdot M_3)$ and $((S\cdot M_1)\cdot M_2)\cdot M_3 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K N$ with its last applied rule $(\mathsf{dpr}_{CL}^{\mathsf{grm}})$. Then the derivation must have the form

$$\dfrac{\dfrac{\dfrac{\dfrac{}{S \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K S}\,(\mathsf{dpr}_{CL}^{\mathsf{grm}}) \quad \begin{array}{c}\vdots\ D_1\\ M_1 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K N_1\end{array}}{S\cdot M_1 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K S\cdot N_1}\,(\mathsf{dpr}_{CL}^{\mathsf{grm}}) \quad \begin{array}{c}\vdots\ D_2\\ M_2 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K N_2\end{array}}{(S\cdot M_1)\cdot M_2 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K (S\cdot N_1)\cdot N_2}\,(\mathsf{dpr}_{CL}^{\mathsf{grm}}) \quad \begin{array}{c}\vdots\ D_3\\ M_3 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K N_3\end{array}}{((S\cdot M_1)\cdot M_2)\cdot M_3 \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K ((S\cdot N_1)\cdot N_2)\cdot N_3\ (\equiv N)}\,(\mathsf{dpr}_{CL}^{\mathsf{grm}})$$

Hence we can construct a derivation of $(M_1\cdot M_3)\cdot(M_2\cdot M_3) \mathrel{-\!\!\shortparallel\!\!\rightarrow}_K (N_1\cdot N_3)\cdot(N_2\cdot N_3)$ from D_1, D_2 and D_3 by using the rule $(\mathsf{dpr}_{CL}^{\mathsf{grm}})$. We also have $((S\cdot N_1)\cdot N_2)\cdot N_3 \longrightarrow_S (N_1\cdot N_3)\cdot(N_2\cdot N_3)$, and so the condition (sc_1) is satisfied. On the other hand, it is seen that the condition (sc_2) is vacuously satisfied.

Next we consider the case where both (ρ_1) and (ρ_2) are the rule (S). Then the condition (sc_1) can be checked similarly to the above case. For the condition (sc_2), we only have to check the case where both redexes are at the root, and in that case the claim clearly holds.

The case where both (ρ_1) and (ρ_2) are the rule (K) can be checked similarly.

Proposition 3.6. $\longrightarrow_S \cup \longrightarrow_K$ *is confluent.*

Proof. By Lemma 3.2(2), Lemma 3.3, Theorem 3.4 and Example 3.5. $\square$

It is possible to show that the conditions (sc_1) and (sc_2) are always satisfied for any two rules of an orthogonal system. The details are omitted.

We generalise Theorem 3.4 on reduction rules (ρ_1) and (ρ_2) to that on sets of reduction rules R_1 and R_2. For a set of reduction rules R, we define $\longrightarrow_R = \bigcup_{(\rho)\in R} \longrightarrow_\rho$. Also, the disjoint parallel R-reduction $\twoheadrightarrow\!\!\!\!\!\!|\!|_R$ is defined by the rule $(\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$ and the rules (dpr_ρ) for all $(\rho) \in R$.

Theorem 3.7 (Sufficient condition for strong commutation)**.** *Let R_1 and R_2 be sets of reduction rules of a system $\mathscr{L}$. Suppose that $\longrightarrow_{R_1}$ and $\twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2}$ satisfy the following conditions:*

(SC_1) *If $M \longrightarrow_{R_1} M'$ with the R_1-redex at the root, and $M \twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2} N$ is derived with $(\mathsf{dpr}^{\mathsf{grm}}_{\mathscr{L}})$ as the last applied rule, then there exists N' such that $M' \twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2} N'$ and $N \longrightarrow^*_{R_1} N'$.*

(SC_2) *For any $(\rho_2) \in R_2$, if $\mathscr{C}^{\mathsf{left}}_{\rho_2}[M_i]_i \longrightarrow_{R_1} M'$ with the R_1-redex not in any M_i, then there exists N' such that $M' \twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2} N'$ and $\mathscr{C}^{\mathsf{right}}_{\rho_2}[M_i]_i \longrightarrow^*_{R_1} N'$.*

Then $\longrightarrow_{R_1}$ strongly commutes with $\twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2}$.

Proof. By induction on the derivation of $M \twoheadrightarrow\!\!\!\!\!\!|\!|_{R_2} N$, similarly to the proof of Theorem 3.4. $\square$

4 Extension to systems with bound variables and conditional rules

The remainder of the paper is concerned with rewriting systems possibly with bound variables and conditional rules. For unconditional rules, the sufficient conditions (sc_1) and (sc_2) (or more generally, (SC_1) and (SC_2)) for strong commutation work as well as in the case without bound variables, except some subtleties caused by identifying α-convertible terms.

We explain the contents using examples from $\lambda\mathsf{x}$-calculus [4].

Definition 4.1 (Grammar of $\lambda\mathsf{x}$)**.** *The set of terms of the $\lambda\mathsf{x}$-calculus is defined by the following grammar:*

$$M,N ::= x \mid \lambda x.M \mid MN \mid M\langle x := N\rangle$$

where x ranges over a denumerable set of variables.

In the following we assume that x, y and z denote distinct variables and that the construct $\_\langle x := \_\rangle$ binds more strongly than $\lambda x.\_$. The notions of free and bound variables are extended from those for λ-terms by the clause that the variable x in $M\langle x := N\rangle$ binds the free occurrences of x in M. The set of free variables occurring in a term M is denoted by $\mathrm{FV}(M)$. We identify α-convertible terms and use $\equiv$ to denote syntactic equality modulo α-conversion.

Example 4.2. For the $\lambda\mathsf{x}$-calculus, $\mathscr{C}^{\mathsf{grm}}_{\lambda\mathsf{x}}[\,]$ is defined by

$$\mathscr{C}^{\mathsf{grm}}_{\lambda\mathsf{x}}[\,] ::= x \mid \lambda x.\square_1 \mid \square_1\square_2 \mid \square_1\langle x := \square_2\rangle$$

Definition 4.3 (Reduction systems of x and x^-)**.** *The reduction rules of the system x are:*

$(\mathsf{x1})$	$M\langle x := N\rangle$	$\to$	M	*if* $x \notin \mathrm{FV}(M)$
$(\mathsf{x2})$	$x\langle x := N\rangle$	$\to$	N	
$(\mathsf{x3})$	$(\lambda y.M)\langle x := N\rangle$	$\to$	$\lambda y.M\langle x := N\rangle$	
$(\mathsf{x4})$	$(M_1M_2)\langle x := N\rangle$	$\to$	$M_1\langle x := N\rangle M_2\langle x := N\rangle$	

The reduction relation $\longrightarrow_{\mathsf{x}}$ *is defined by the contextual closure of the rules* (x1)-(x4)*. The system* x^- *consists of the rules* (x2)-(x4) *and the following:*

$$(\mathsf{x1}^-) \qquad y\langle x := N\rangle \;\rightarrow\; y$$

The reduction relation $\longrightarrow_{\mathsf{x}^-}$ *is defined by the contextual closure of the rules* (x2)-(x4) *and* ($\mathsf{x1}^-$).

As usual in considering terms up to α-equivalence, we rename bound variables in a redex if accidental capture of free variables after reduction occurs. Thus, for example,

$$(\lambda y.yx)\langle x := y\rangle \equiv (\lambda z.zx)\langle x := y\rangle \longrightarrow_{\mathsf{x3}} \lambda z.(zx)\langle x := y\rangle \text{ and not } (\lambda y.yx)\langle x := y\rangle \longrightarrow_{\mathsf{x3}} \lambda y.(yx)\langle x := y\rangle$$

However, renaming of variables causes a subtlety in our contextual notation of reduction rule schema. We look into the problem below.

Example 4.4. For the rules of the system x^-,

$$\begin{array}{lcl}
\mathscr{C}^{\mathsf{left}}_{\mathsf{x2}}[\,] \equiv x\langle x := \square_1\rangle & \text{and} & \mathscr{C}^{\mathsf{right}}_{\mathsf{x2}}[\,] \equiv \square_1 \\
\mathscr{C}^{\mathsf{left}}_{\mathsf{x3}}[\,] \equiv (\lambda y.\square_1)\langle x := \square_2\rangle & \text{and} & \mathscr{C}^{\mathsf{right}}_{\mathsf{x3}}[\,] \equiv \lambda y.\square_1\langle x := \square_2\rangle \\
\mathscr{C}^{\mathsf{left}}_{\mathsf{x4}}[\,] \equiv (\square_1\square_2)\langle x := \square_3\rangle & \text{and} & \mathscr{C}^{\mathsf{right}}_{\mathsf{x4}}[\,] \equiv \square_1\langle x := \square_3\rangle\square_2\langle x := \square_3\rangle \\
\mathscr{C}^{\mathsf{left}}_{\mathsf{x1}^-}[\,] \equiv y\langle x := \square_1\rangle & \text{and} & \mathscr{C}^{\mathsf{right}}_{\mathsf{x1}^-}[\,] \equiv y
\end{array}$$

Note that each context above is parametrised by variables. So we suppose that the notations $\mathscr{C}^{\mathsf{left}}_{\rho}[\,]$ and $\mathscr{C}^{\mathsf{right}}_{\rho}[\,]$ for each reduction rule schema are indexed with a sequence of the bound variables whose scope includes one of the holes. Then the above reduction step by the rule (x3) is written as follows:

$$\mathscr{C}^{\mathsf{left}}_{\mathsf{x3},y,x}[yx,y] \equiv \mathscr{C}^{\mathsf{left}}_{\mathsf{x3},z,x}[zx,y] \longrightarrow_{\mathsf{x3}} \mathscr{C}^{\mathsf{right}}_{\mathsf{x3},z,x}[zx,y]$$

In fact, when considering terms up to α-equivalence, one always has to pay attention to such kind of notation ranging over contexts, since free variables in terms with which holes are filled may be renamed by α-conversion. This remark applies to the proof of Theorem 3.4 in the presence of bound variables, where we use the fact that properties on reduction do not depend on which names of free variables are employed.

Now it is not difficult to check the conditions (sc_1) and (sc_2) for the rules of the system x^-.

Example 4.5. Let (ρ_1) and (ρ_2) be the rules (x4) and ($\mathsf{x1}^-$), respectively. To check the condition (sc_1), suppose $(M_1M_2)\langle x := M_3\rangle \longrightarrow_{\mathsf{x4}} M_1\langle x := M_3\rangle M_2\langle x := M_3\rangle$ and $(M_1M_2)\langle x := M_3\rangle \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N$ with its last applied rule ($\mathsf{dpr}^{\mathsf{grm}}_{\lambda\mathsf{x}}$). Then the derivation must have the form

$$\dfrac{\dfrac{\dfrac{\vdots\, D_1}{M_1 \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N_1} \quad \dfrac{\vdots\, D_2}{M_2 \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N_2}}{M_1M_2 \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N_1N_2}\,(\mathsf{dpr}^{\mathsf{grm}}_{\lambda\mathsf{x}}) \quad \dfrac{\vdots\, D_3}{M_3 \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N_3}}{(M_1M_2)\langle x := M_3\rangle \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} (N_1N_2)\langle x := N_3\rangle\ (\equiv N)}\,(\mathsf{dpr}^{\mathsf{grm}}_{\lambda\mathsf{x}})$$

Hence we can construct a derivation of $M_1\langle x := M_3\rangle M_2\langle x := M_3\rangle \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}^-} N_1\langle x := N_3\rangle N_2\langle x := N_3\rangle$ from D_1, D_2 and D_3 by using the rule ($\mathsf{dpr}^{\mathsf{grm}}_{\lambda\mathsf{x}}$). We also have $(N_1N_2)\langle x := N_3\rangle \longrightarrow_{\mathsf{x4}} N_1\langle x := N_3\rangle N_2\langle x := N_3\rangle$, and so the condition ($\mathsf{sc}_1$) is satisfied. On the other hand, it is seen that the condition (sc_2) is vacuously satisfied.

Next we consider the case where both (ρ_1) and (ρ_2) are the rule (x4). Then the condition (sc_1) can be checked similarly to the above case. For the condition (sc_2), we only have to check the case where both redexes are at the root, and in that case the claim clearly holds.

For the other combinations of rules, the conditions can be checked similarly.

Proposition 4.6. $\longrightarrow_{\mathsf{x}^-}$ *is confluent.*

Proof. By Lemma 3.2(2), Lemma 3.3, Theorem 3.4 and Example 4.5. □

For conditional rules like $(\mathsf{x1})$ of the system x, the notion of reduction rule schema of a rule (ρ) is extended from the pair $\langle \mathscr{C}_\rho^{\mathsf{left}}[\,], \mathscr{C}_\rho^{\mathsf{right}}[\,]\rangle$ to the triple with $\mathscr{C}_\rho^{\mathsf{cond}}[\,]$, which is obtained by replacing, in the conditional part of the rule (ρ), each meta-variable by the respective hole.

Example 4.7. For the rule $(\mathsf{x1})$ of the system x,

$$\mathscr{C}_{\mathsf{x1},x}^{\mathsf{left}}[\,] \equiv \square_1\langle x := \square_2\rangle \quad \text{and} \quad \mathscr{C}_{\mathsf{x1},x}^{\mathsf{right}}[\,] \equiv \square_1 \quad \text{and} \quad \mathscr{C}_{\mathsf{x1},x}^{\mathsf{cond}}[\,] \equiv x \notin \mathrm{FV}(\square_1)$$

We modify the rule (dpr_ρ) in Definition 2.8 so that it can be applied only when $\mathscr{C}_\rho^{\mathsf{cond}}[\![M_i]\!]_i$ holds, which means the expression $\mathscr{C}_\rho^{\mathsf{cond}}[M_i]_i$ is evaluated to true. Then Lemmas 2.9 and 2.10 still hold. Moreover, by adding a new condition (stb) and modifying (sc_2), the claim corresponding to Theorem 3.4 has the following form. (Adding (stb) can be seen as an application of stability [2] to strong commutation.)

Theorem 4.8 (Sufficient condition for strong commutation)**.** *Let (ρ_1) and (ρ_2) be reduction rules of a system $\mathscr{L}$. Suppose that $\longrightarrow_{\rho_1}$ and $\twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2}$ satisfy the following conditions:*

- (sc_1) *If $\mathscr{C}_{\rho_1}^{\mathsf{left}}[M_i]_i \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N$ is derived with $(\mathsf{dpr}_{\mathscr{L}}^{\mathsf{grm}})$ as the last applied rule, and $\mathscr{C}_{\rho_1}^{\mathsf{cond}}[\![M_i]\!]_i$ holds, then there exists N' such that $\mathscr{C}_{\rho_1}^{\mathsf{right}}[M_i]_i \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N'$ and $N \longrightarrow_{\rho_1}^* N'$. (This is substantially the same as (sc_1) in Theorem 3.4.)*
- (sc_2) *If $\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_i]_i \longrightarrow_{\rho_1} M'$ with the ρ_1-redex not in any M_i, and $\mathscr{C}_{\rho_2}^{\mathsf{cond}}[\![M_i]\!]_i$ holds, then there exists N' such that $M' \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N'$ and $\mathscr{C}_{\rho_2}^{\mathsf{right}}[M_i]_i \longrightarrow_{\rho_1}^* N'$.*
- (stb) *If $\mathscr{C}_{\rho_2}^{\mathsf{cond}}[\![M_1,\dots,M_n]\!]$ holds and $M_i \longrightarrow_{\rho_1} M_i'$ for some $i \in \{1,\dots,n\}$, then $\mathscr{C}_{\rho_2}^{\mathsf{cond}}[\![M_1,\dots,M_{i-1},M_i',M_{i+1},\dots,M_n]\!]$ also holds.*

Then $\longrightarrow_{\rho_1}$ strongly commutes with $\twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2}$.

Proof. By induction on the derivation of $M \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N$, similarly to the proof of Theorem 3.4. The case where the last applied rule in the derivation of $M \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N$ is $(\mathsf{dpr}_{\mathscr{L}}^{\mathsf{grm}})$ is proved in the same way as in the proof of Theorem 3.4. So suppose that $M \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} N$ is derived by the rule (dpr_{ρ_2}) of the form

$$\frac{}{\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\dots,M_n] \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\dots,M_n]}\ (\mathsf{dpr}_{\rho_2})$$

where $\mathscr{C}_{\rho_2}^{\mathsf{cond}}[\![M_1,\dots,M_n]\!]$ holds. First we consider the case where the reduction $M \longrightarrow_{\rho_1} M'$ takes place in $\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\dots,M_n]$ with $M_i \longrightarrow_{\rho_1} M_i'$ for some $i \in \{1,\dots,n\}$. Then by the condition (stb), $\mathscr{C}_{\rho_2}^{\mathsf{cond}}[\![M_1,\dots,M_i',\dots,M_n]\!]$ holds, and so by the rule (dpr_{ρ_2}), we have

$$M' \equiv \mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\dots,M_i',\dots,M_n] \twoheadrightarrow\!\!\!\!\!\mapsto_{\rho_2} \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\dots,M_i',\dots,M_n]$$

Also, we have

$$N \equiv \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\dots,M_i,\dots,M_n] \longrightarrow_{\rho_1}^* \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\dots,M_i',\dots,M_n]$$

Hence the claim follows by taking $N' \equiv \mathscr{C}_{\rho_2}^{\mathsf{right}}[M_1,\dots,M_i',\dots,M_n]$. The case where the redex of $M \longrightarrow_{\rho_1} M'$ is not in any M_i of $\mathscr{C}_{\rho_2}^{\mathsf{left}}[M_1,\dots,M_n]$ follows from the condition (sc_2). □

Now we can check the conditions (sc_1), (sc_2) and (stb) for the rule $(\mathsf{x1})$ and the other rules of the system x.

Example 4.9. Let (ρ_1) and (ρ_2) be the rules $(\mathsf{x4})$ and $(\mathsf{x1})$, respectively. Then the condition (sc_1) can be checked similarly to the case of $(\mathsf{x4})$ and $(\mathsf{x1}^-)$ in Example 4.5. To check the condition (sc_2), we have to consider this time the case where $\mathscr{C}^{\mathsf{left}}_{\mathsf{x1},x}[M_1M_2,M_3] \equiv (M_1M_2)\langle x := M_3\rangle \longrightarrow_{\mathsf{x4}} M_1\langle x := M_3\rangle M_2\langle x := M_3\rangle$ and $\mathscr{C}^{\mathsf{cond}}_{\mathsf{x1},x}[M_1M_2,M_3] \equiv x \notin \mathrm{FV}(M_1M_2)$. Then we have $M_1\langle x := M_3\rangle M_2\langle x := M_3\rangle \twoheadrightarrow\!\!\!\!\!+_{\mathsf{x1}} M_1M_2$ and $\mathscr{C}^{\mathsf{right}}_{\mathsf{x1},x}[M_1M_2,M_3] \equiv M_1M_2$. Hence the condition (sc_2) is satisfied. Also, since $\mathscr{C}^{\mathsf{cond}}_{\mathsf{x1},x}[\![M_1,M_2]\!]$ (i.e. $x \notin \mathrm{FV}(M_1)$ is true) and $M_1 \longrightarrow_{\mathsf{x4}} M_1'$ imply $\mathscr{C}^{\mathsf{cond}}_{\mathsf{x1},x}[\![M_1',M_2]\!]$ (i.e. $x \notin \mathrm{FV}(M_1')$ is true), the condition (stb) is satisfied.

For the cases where (ρ_1) and (ρ_2) are the rules $(\mathsf{x2})$ and $(\mathsf{x1})$ or the rules $(\mathsf{x3})$ and $(\mathsf{x1})$, the conditions can be checked similarly. For the case where both (ρ_1) and (ρ_2) are the rule $(\mathsf{x1})$, the conditions can be easily checked.

Proposition 4.10. $\longrightarrow_{\mathsf{x}}$ *is confluent.*

Proof. We have already checked in Example 4.5 the conditions (sc_1) and (sc_2) for all combinations of the unconditional rules $(\mathsf{x2})$-$(\mathsf{x4})$, and checked in Example 4.9 the conditions (sc_1), (sc_2) and (stb) for the rule $(\mathsf{x1})$ and the other rules. Hence by Lemma 3.2(2), Lemma 3.3 and Theorem 4.8, we conclude that $\longrightarrow_{\mathsf{x}}$ is confluent. □

It is sometimes useful to work with (sc_1) and (sc_2) rather than general versions (SC_1) and (SC_2) with larger sets of rules, since there are more possibilities of reusing already obtained results on commutation of part of the rules of the system, as seen in the proof of the above proposition.

5 Conclusion

We proposed a specification format of rewriting systems and presented sufficient conditions for strong commutation to prove confluence for classes of rewriting systems possibly with bound variables and conditional rules. We also pointed out subtleties that arise from identifying α-convertible terms in developing a general framework of proving confluence for rewriting systems with bound variables.

As remarked after Example 4.4, one has to pay attention to notations ranging over contexts when identifying α-convertible terms. This problem appears even if one employs a specification format of rewriting systems with meta-variables and capturing substitution (or assignment) as in [17, 6, 11]. In that case, the substitution should not be considered separately from the terms to which it is applied. In the framework of CCERSs [6, 11], the problem has been treated to some extent by requiring closure of admissibility under the renaming of bound variables (cf. footnote 6 of [11, p. 124]). However, it is not sufficient for dealing with the relation between a pattern (which corresponds to $\mathscr{C}^{\mathsf{left}}_{\rho}[\,]$ in our notation) and arguments (i.e. M_i's in $\mathscr{C}^{\mathsf{left}}_{\rho}[M_i]_i$).

In the framework of nominal rewriting [5], where α-convertible terms are not identified, the above problem can be treated rigorously (as far as conditions of conditional rules are restricted to freshness ones). In particular, the notions of uniformity [5] and α-stability [16] stipulate suitable conditions for the rules concerned. Confluence criteria using strong commutation or parallel critical pairs [7, 18] have not yet been proposed in the framework of nominal rewriting.

On the other hand, traditional higher-order rewriting frameworks [12, 13] use higher-order terms involving some meta-level calculus (e.g. the simply-typed λ-calculus). In those cases, one has to first transform reduction rules into rules of a higher-order system. (An example of transformation is found in [10, p. 145].) Such a process is rarely written down in the literature, though it is not completely straightforward.

Acknowledgements I am grateful to the anonymous referees for valuable comments. This work was supported by JSPS KAKENHI Grant Numbers JP16K00091 and JP17K00005.

References

[1] Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press (1998)

[2] Bergstra, J.A., Klop, J.W.: Conditional rewrite rules: confluence and termination. Journal of Computer and System Sciences 32(3), 323–362 (1986)

[3] Blanqui, F., Kirchner, C., Riba, C.: On the confluence of lambda-calculus with conditional rewriting. Theoretical Computer Science 411(37), 3301–3327 (2010)

[4] Bloo, R., Rose, K.H.: Preservation of strong normalisation in named lambda calculi with explicit substitution and garbage collection. In: Proceedings of Computing Science in the Netherlands (CSN 1995). pp. 62–72. (1995)

[5] Fernández, M., Gabbay, M.J.: Nominal rewriting. Information and Computation 205(6), 917–965 (2007)

[6] Glauert, J.R.W., Kesner, D., Khasidashvili, Z.: Expression reduction systems and extensions: An overview. In: Processes, Terms and Cycles. LNCS, vol. 3838, pp. 496–553. Springer-Verlag (2005)

[7] Gramlich, B.: Confluence without termination via parallel critical pairs. In: Proceedings of the 21st Colloquium on Trees in Algebra and Programming (CAAP 1996). LNCS, vol. 1059, pp. 211–225. Springer-Verlag (1996)

[8] Hindley, J.R.: The Church-Rosser Property and a Result in Combinatory Logic. Ph.D. thesis, University of Newcastle-upon-Tyne (1964)

[9] Huet, G.: Confluent reductions: abstract properties and applications to term rewriting systems. Journal of the Association for Computing Machinery 27(4), 797–821 (1980)

[10] Joachimski, F., Matthes, R.: Standardization and confluence for a lambda calculus with generalized applications. In: Proceedings of the 11th International Conference on Rewriting Technique and Applications (RTA 2000). LNCS, vol. 1833, pp. 141–155. Springer-Verlag (2000)

[11] Khasidashvili, Z., Ogawa, M., van Oostrom, V.: Perpetuality and uniform normalization in orthogonal rewrite systems. Information and Computation 164(1), 118–151 (2001)

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

[13] Mayr, R., Nipkow, T.: Higher-order rewrite systems and their confluence. Theoretical Computer Science 192(1), 3–29 (1998)

[14] de Moura, F.L.C., Kesner, D., Ayala-Rincón, M.: Metaconfluence of calculi with explicit substitutions at a distance. In: Proceedings of the 34th International Conference on Foundation of Software Technology and Theoretical Computer Science (FSTTCS 2014). LIPIcs, vol. 29, pp. 391–402. (2014)

[15] Shintani, K., Hirokawa, N.: CoLL: a confluence tool for left-linear term rewrite systems. In: Proceedings of the 25th International Conference on Automated Deduction (CADE 2015). LNCS, vol. 9195, pp. 127–136. Springer-Verlag (2015)

[16] Suzuki, T., Kikuchi, K., Aoto, T., Toyama, Y.: Confluence of orthogonal nominal rewriting systems revisited. In: Proceedings of the 26th International Conference on Rewriting Technique and Applications (RTA 2015). LIPIcs, vol. 36, pp. 301–317. (2015)

[17] Takahashi, M.: λ-calculi with conditional rules. In: Proceedings of the 1st International Conference on Typed Lambda Calculi and Applications. LNCS, vol. 664, pp. 406–417. Springer-Verlag (1993)

[18] Toyama, Y.: On the Church-Rosser property of term rewriting systems. Technical Report 17672, NTT ECL (1981), *In Japanese*