# Level-Confluence of 3-CTRSs in Isabelle/HOL*

Christian Sternagel and Thomas Sternagel

University of Innsbruck, Innsbruck, Austria
{christian.sternagel, thomas.sternagel}@uibk.ac.at

**Abstract**

We present an Isabelle/HOL formalization of an earlier result by Suzuki, Middeldorp, and Ida; namely that a certain class of conditional rewrite systems is level-confluent. Our formalization is basically along the lines of the original proof, from which we deviate mostly in the level of detail as well as concerning some basic definitions.

## 1 Introduction

In the realm of standard term rewriting, many properties of term rewrite systems (TRSs) can be conveniently checked "at the push of a button" due to a wealth of existing automated tools. To maximize the reliability of this approach, such automated tools are progressively complemented by certifiers, that is, verified programs that rigorously ensure that the output of an automated tool for a given input is correct. At the time of writing the prevalent methodology for certifier development consists of the following two phases: First, employ a proof assistant (in our case Isabelle/HOL [5]) in order to formalize the underlying theory, resulting in a *formal library* (in our case IsaFoR,[1] an *Isabelle/HOL Formalization of Rewriting*). Then, verify a program using this library, resulting in the actual certifier (in our case CeTA [9]).

Our ultimate goal is to establish the same state of the art also for conditional term rewrite systems (CTRSs). As a starting point, we present our Isabelle/HOL formalization of the following result:

**Lemma 1** (Suzuki et al. [8, Corollary 4.7])**.** *Orthogonal, properly oriented, right-stable 3-CTRSs are level-confluent.* □

Which is actually a corollary of a more general result, whose statement – together with a high-level overview of its proof – we defer until after we have established some necessary preliminaries.

The development we describe in this note is now part of the IsaFoR library and is freely available for inspection at:

http://cl2-informatik.uibk.ac.at/rewriting/mercurial.cgi/IsaFoR/file/a2cd778de34a/IsaFoR/Conditional_Rewriting/Level_Confluence.thy

Throughout the remainder we will from time to time refer to the Isabelle/HOL sources of our development (by active hyperlink).

*The research described in this paper is supported by FWF (Austrian Science Fund) project P27502.

[1] http://cl-informatik.uibk.ac.at/software/ceta/

## 2 Preliminaries

We assume familiarity with (conditional) term rewriting [1, 6]. In the sequel we consider oriented 3-CTRSs where extra variables in conditions and right-hand sides of rewrite rules are allowed, i.e., for all rules $\ell \to r \Leftarrow c$ in the CTRS we only demand $\mathcal{V}\mathsf{ar}(r) \subseteq \mathcal{V}\mathsf{ar}(\ell, c)$. For such systems extended TRSs $\mathcal{R}_n$ are inductively defined for each level $n \geqslant 0$ as follows

$$\mathcal{R}_0 = \varnothing$$
$$\mathcal{R}_{n+1} = \{\ell\sigma \to r\sigma \mid \ell \to r \Leftarrow c \in \mathcal{R} \text{ and } s\sigma \xrightarrow[\mathcal{R}_n]{*} t\sigma \text{ for all } s \approx t \in c\}$$

where $\to_{\mathcal{R}_n}$ denotes the standard (unconditional) rewrite relation of the TRS $\mathcal{R}_n$. We write $s \to_{\mathcal{R}} t$ if we have $s \to_{\mathcal{R}_n} t$ for some $n \geqslant 0$. Moreover, for brevity, the latter is written $s \to_n t$ whenever the corresponding CTRS is clear from the context. Given two variable disjoint variants $\ell_1 \to r_1 \Leftarrow c_1$ and $\ell_2 \to r_2 \Leftarrow c_2$ of rules in a CTRS $\mathcal{R}$, a function position $p$ in $\ell_1$, and a most general unifier (mgu) $\mu$ of $\ell_1|_p$ and $\ell_2$; we call the triple $(\ell_1 \to r_1 \Leftarrow c_1, \ell_2 \to r_2 \Leftarrow c_2, p)$ a *conditional overlap* of $\mathcal{R}$. A conditional overlap $(\ell_1 \to r_1 \Leftarrow c_1, \ell_2 \to r_2 \Leftarrow c_2, p)$ with mgu $\mu$ is *infeasible* (that is, cannot occur during actual rewriting) if there is no substitution $\sigma$ such that $s\sigma \to_{\mathcal{R}}^* t\sigma$ for all $s \approx t$ in $c_1\mu, c_2\mu$.

*A note on permutations.* At the highly formal level of Isabelle/HOL (which we tend to avoid in the following exposition) we employ an existing formalization of *permutation types* (that is, types that contain variables which may be renamed w.r.t. a given permutation) to tackle variable renamings, renaming rules apart, and checking whether two rules are variants of each other. This abstract view on renamings (as opposed to explicit renaming functions on strings) proved to be useful in previous applications [3, 4].

We call a CTRS *almost orthogonal* [2] *(modulo infeasibility)* if it is left-linear and all its conditional overlaps are either infeasible or take place at root position ($\ell_1\mu = \ell_2\mu$) and are either between variants of the same rule or also result in syntactically equal right-hand sides ($r_1\mu = r_2\mu$). A CTRS $\mathcal{R}$ is called *properly oriented* if for all rules $\ell \to r \Leftarrow s_1 \approx t_1, \ldots, s_k \approx t_k \in \mathcal{R}$ where $\mathcal{V}\mathsf{ar}(r) \nsubseteq \mathcal{V}\mathsf{ar}(\ell)$ and $1 \leqslant i \leqslant k$ we have $\mathcal{V}\mathsf{ar}(s_i) \subseteq \mathcal{V}\mathsf{ar}(\ell, t_1, \ldots, t_{i-1})$. It is called *right-stable* if for every rule $\ell \to r \Leftarrow s_1 \approx t_1, \ldots, s_k \approx t_k \in \mathcal{R}$ and $1 \leqslant i \leqslant k$ we have $\mathcal{V}\mathsf{ar}(\ell, s_1, \ldots, s_i, t_1, \ldots, t_{i-1}) \cap \mathcal{V}\mathsf{ar}(t_i) = \varnothing$ and $t_i$ is either a linear constructor term or a ground $\mathcal{R}_\mathsf{u}$-normal form.

We say that two binary relations $\to_\alpha$ and $\to_\beta$ have the *commuting diamond property* [1], whenever ${}_\alpha\!\leftarrow \cdot \to_\beta \subseteq \to_\beta \cdot {}_\alpha\!\leftarrow$. Moreover, we adopt the notion of *extended parallel rewriting* from Suzuki et al. [8].

**Definition 2.** Let $\mathcal{R}$ be a CTRS. We say that there is an *extended parallel $\mathcal{R}$-rewrite step at level* $n$ from $s$ to $t$, written $s \mathrel{+\!\!\!\!\Vdash\!\!\to}_{\mathcal{R}_n} t$ (or $s \mathrel{+\!\!\!\!\Vdash\!\!\to}_n t$ for brevity), whenever we have a multihole context $C$, and sequences of terms $s_1, \ldots, s_k$ and $t_1, \ldots, t_k$, such that $s = C[s_1, \ldots, s_k]$, $t = C[t_1, \ldots, t_k]$, and for all $1 \leqslant i \leqslant k$ we have one of $(s_i, t_i) \in \mathcal{R}_n$ (that is, a root-step at level $n$) and $s_i \to_{n-1}^* t_i$.

Suzuki et al. [8], state this definition slightly differently, that is, instead of multihole contexts they try to rely exclusively on sets of positions:

> *We write $s \mathrel{+\!\!\!\!\Vdash\!\!\to}_n t$ if there exists a subset $P$ of pairwise disjoint positions in $s$ such that for all $p \in P$ either $(s|_p, t|_p) \in \mathcal{R}_n$ or $s|_p \to_{n-1}^* t|_p$.*

While it is quite clear what is meant, a slight problem (at least for a formal development inside a proof assistant) is the fact that this definition does not enforce $t$ to be exactly the same as $s$

outside of the positions in $P$, that is, it does not require the multihole context around the $|P|$ rewrite sequences to stay the same. In order to express this properly, it seems unavoidable to employ multihole contexts (or something equivalent).

In the remainder we employ the convention that the number of holes of a multihole context, is denoted by the corresponding lower-case letter, e.g., $c$ for a multihole context $C$, $d$ for $D$, $e$ for $E$, etc.

# 3 The Main Result

As remarked in the last two sections of Suzuki et al. [8], we actually consider *almost orthogonal* systems modulo *infeasibility*. We are now in a position to state the main theorem.

**Theorem 3** (Suzuki et al. [8, Theorem 4.6])**.** *Let $\mathcal{R}$ be an almost orthogonal (modulo infeasibility), properly oriented, right-stable 3-CTRS. Then, for any two levels $m$ and $n$, the extended parallel rewrite relations $\mathrel{\Vert\!\!\!\rightarrowtail}_m$ and $\mathrel{\Vert\!\!\!\rightarrowtail}_n$, have the commuting diamond property.*

As a special case of the above theorem, we obtain that for a fixed level $n$, the relation $\mathrel{\Vert\!\!\!\rightarrowtail}_n$ has the diamond property. Moreover, it is well known that whenever a relation $S$ with the diamond property, is between a relation $R$ and its reflexive, transitive closure (that is, $R \subseteq S \subseteq R^*$), then $R$ is confluent. Taken together, this yields level-confluence of $\rightarrow_{\mathcal{R}}$, since clearly $\rightarrow_n \subseteq \mathrel{\Vert\!\!\!\rightarrowtail}_n \subseteq \rightarrow_n^*$.

We now give a high-level overview of the proof of Theorem 3. The general structure is similar to the one followed by Suzuki et al. [8], only that we employ multihole contexts instead of sets of positions. Therefore, we do not give all the details (if you are interested, see `Conditional_Rewriting/Level_Confluence`, starting from `comm_epar_n` in line 1499), but mostly comment on the parts that differ (if only slightly).

*Proof (Sketch) of Theorem 3.* We proceed by complete induction on $m + n$. By induction hypothesis (IH) we may assume the result for all $m' + n' < m + n$. Now consider the peak $t \;{}_m\!\mathrel{\leftarrowtail\!\!\!\Vert}\; s \mathrel{\Vert\!\!\!\rightarrowtail}_n u$. If any of $m$ and $n$ equals 0, we are done (since $\mathrel{\Vert\!\!\!\rightarrowtail}_0$ is the identity relation). Thus we may assume $m = m' + 1$ and $n = n' + 1$ for some $m'$ and $n'$. By the definition of extended parallel rewriting, we obtain multihole contexts $C$ and $D$, and sequences of terms $s_1, \ldots, s_c$, $t_1, \ldots, t_c$, $u_1, \ldots, u_d$, $v_1, \ldots, v_d$, such that $s = C[s_1, \ldots, s_c]$ and $t = C[t_1, \ldots, t_c]$, as well as $s = D[u_1, \ldots, u_d]$ and $u = D[v_1, \ldots, v_d]$; and $(s_i, t_i) \in \mathcal{R}_m$ or $s_i \rightarrow_{m'}^* t_i$ for all $1 \leqslant i \leqslant c$, as well as $(u_i, v_i) \in \mathcal{R}_n$ or $u_i \rightarrow_{n'}^* v_i$ for all $1 \leqslant i \leqslant d$.

Now we identify the common part $E$ of $C$ and $D$, employing the semi-lattice properties of multihole contexts (see `Rewriting/Multihole_Context`), that is, $E = C \sqcap D$. Then $C = E[C_1, \ldots, C_e]$ and $D = E[D_1, \ldots, D_e]$ for some multihole contexts $C_1, \ldots, C_e$ and $D_1, \ldots, D_e$ such that for each $1 \leqslant i \leqslant e$ we have $C_i = \square$ or $D_i = \square$. This also means that there is a sequence of terms $s_1', \ldots, s_e'$ such that $s = E[s_1', \ldots, s_e']$ and for all $1 \leqslant i \leqslant e$, we have $s_i' = C_i[s_{k_i}, \ldots, s_{k_i + c_i - 1}]$ for some subsequence $s_{k_i}, \ldots, s_{k_i + c_i - 1}$ of $s_1, \ldots, s_c$ (we denote similar terms for $t$, $u$, and $v$ by $t_i'$, $u_i'$, and $v_i'$, respectively). Moreover, note that by construction $s_i' = u_i'$ for all $1 \leqslant i \leqslant e$. Since extended parallel rewriting is closed under multihole contexts (see `epar_n_mctxt`), it suffices to show that for each $1 \leqslant i \leqslant e$ there is a term $v$ such that $t_i' \mathrel{\Vert\!\!\!\rightarrowtail}_n v \;{}_m\!\mathrel{\leftarrowtail\!\!\!\Vert}\; v_i'$, in order to conclude the proof. We concentrate on the case that $C_i = \square$ (the case $D_i = \square$ is completely symmetric). Moreover, note that when we have $s_i' \rightarrow_{m'}^* t_i'$, the proof concludes by IH (together with some basic properties of the involved relations), and thus we remain with $(s_i', t_i') \in \mathcal{R}_m$. At this point we distinguish the following cases:

1. ($D_i = \square$). Also here, the non-root case $u'_i \rightarrow^*_{n'} v'_i$ is covered by the IH. Thus, we may restrict to $(u'_i, v'_i) \in \mathcal{R}_n$, giving rise to a root overlap. Since $\mathcal{R}$ is almost orthogonal (modulo infeasibility), this means that either the resulting conditions are not satisfiable or the resulting terms are the same (in both of these cases we are done) or two variable disjoint variants of the same rule $\ell \rightarrow r \Leftarrow c$ were involved. Without extra variables in $r$, this is the end of the story; but since we also want to cover the case where $\mathcal{V}\mathsf{ar}(r) \not\subseteq \mathcal{V}\mathsf{ar}(l)$, we have to reason why this does not cause any trouble. This case is finished by a technical lemma (see `trs_n_peak`) that shows, by induction on the number of conditions in $c$, that we can join the two respective instances of the right-hand side $r$ by extended parallel rewriting. (This is also where proper orientedness and right-stability of $\mathcal{R}$ is first used, that is, were we to relax this properties, we had to adapt the technical lemma.)
2. ($D_i \neq \square$). Then for some $1 \leqslant k \leqslant d$, we have $(u_j, v_j) \in \mathcal{R}_n$ or $u_j \rightarrow^*_{n'} v_j$ for all $k \leqslant j \leqslant k + d_i - 1$, that is, an extended parallel rewrite step of level $n$ from $s'_i = u'_i = D_i[u_{k_i}, \ldots, u_{k_i+d_i-1}]$ to $D_i[v_{k_i}, \ldots, v_{k_i+d_i-1}] = v'_i$. Since $\mathcal{R}$ is almost orthogonal (modulo infeasibility) and, by $D_i \neq \square$, root overlaps are excluded, the constituent parts of the extended parallel step from $s'_i$ to $v'_i$ take place exclusively inside the substitution of the root-step to the left (which is somewhat obvious – as also stated by Suzuki et al. [8] – but surprisingly hard to formalize, see `epar_n_varpeak'`, even more so when having to deal with infeasibility). We again close this case by induction on the number of conditions making use of proper orientedness and right-stability of $\mathcal{R}$, see `epar_n_varpeak` for details. □

## 4 Conclusions and Future Work

In the original paper [8] the proof of Theorem 3 begins after only three definitions (proper orientedness, right-stability, and extended parallel rewriting) and stretches across two pages, including two figures.

By contrast, in our formalization we need 8 definitions and 42 lemmas (mainly stating properties of extended parallel rewriting) before we can start with the main proof. Furthermore, we need two auxiliary technical lemmas to cover the induction proofs on the number of conditions which are nested inside the main case analysis. All in all, resulting in a theory file of about 1500 lines. This yields a *de Bruijn factor* of approximately 18, that is, for every line in the original "paper proof," our formal proof development contains 18 lines of Isar (the formal language of Isabelle/HOL).

In the latest version of our formalization we further relaxed the condition for conditional overlaps to be infeasible (making the result applicable to a larger class of systems) and proved that the main result still holds. More concretely, a conditional overlap $(\ell_1 \rightarrow r_1 \Leftarrow c_1, \ell_2 \rightarrow r_2 \Leftarrow c_2, p)$ with mgu $\mu$ is infeasible iff

$$\forall m\, n.\, \xleftarrow[m]{*} \cdot \xrightarrow[n]{*} \subseteq \xrightarrow[n]{*} \cdot \xleftarrow[m]{*} \implies \nexists \sigma.\, (\forall s \approx t \in c_1\mu.\, s\sigma \xrightarrow[m]{*} t\sigma) \wedge (\forall s \approx t \in c_2\mu.\, s\sigma \xrightarrow[n]{*} t\sigma).$$

That is, we may assume "level-commutation" (which is called shallow-confluence in the literature) when showing that the combined conditions of two rules are not satisfiable. This may be helpful, since it allows us to turn diverging sequences (as would for example result from two conditions with identical left-hand sides) into joining sequences.

**Future Work.** The ultimate goal of this formalization is of course to certify level-confluence proofs of conditional confluence tools, e.g. ConCon [7]. To this end we need executable check functions for the syntactic properties a CTRS has to meet in order to apply the theorem. The check functions for proper orientedness as well as right-stability should be straightforward to implement. For orthogonality, however, there is a small obstacle to overcome. On the one hand, in our formalization we use the abstract notion of *permutation types* inside the definition of conditional critical pairs, only demanding that the set of variables is infinite. While this guarantees that we can always rename two finite sets of variables apart, we do not directly have an executable renaming function at our disposal. On the other hand, in the current version of IsaFoR the type of variables in (standard) critical pairs is fixed to strings, and their definition employs a concrete, executable renaming function. Therefore, it remains to establish a suitable connection between the executable implementation using strings and the abstract definition: for each critical pair in the abstract definition, there is some variant that we obtain by the executable implementation.

Moreover, Suzuki et al. [8] additionally remark (without proof) that the proof of Theorem 3 could easily be adapted to extended proper orientedness. To us, it is not immediately clear how to adapt our formalization. For the time being, we leave this enhancement as future work.

# References

[1] Franz Baader and Tobias Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.

[2] Michael Hanus. On extra variables in (equational) logic programming. In *Proceedings of the 12th International Conference on Logic Programming*, pages 665–679. MIT Press, 1995.

[3] Nao Hirokawa, Aart Middeldorp, and Christian Sternagel. A new and formalized proof of abstract completion. In *Proceedings of the 5th International Conference on Interactive Theorem Proving*, volume 8558 of *Lecture Notes in Computer Science*, pages 292–307. Springer, 2014. doi:10.1007/978-3-319-08970-6_19.

[4] Nao Hirokawa, Aart Middeldorp, and Christian Sternagel. Normalization equivalence of rewrite systems. In *Proceedings of the 3rd International Workshop on Confluence*, 2014.

[5] Tobias Nipkow, Lawrence Charles Paulson, and Makarius Wenzel. *Isabelle/HOL - A Proof Assistant for Higher-Order Logic*, volume 2283 of *Lecture Notes in Computer Science*. Springer, 2002. doi:10.1007/3-540-45949-9.

[6] Enno Ohlebusch. *Advanced Topics in Term Rewriting*. Springer, 2002.

[7] Thomas Sternagel and Aart Middeldorp. Conditional confluence (system description). In *Proceedings of the Joint 25th International Conference on Rewriting Techniques and Applications and 12th International Conference on Typed Lambda Calculi and Applications*, volume 8560 of *Lecture Notes in Computer Science*, pages 456–465. Springer, 2014. doi:10.1007/978-3-319-08918-8_31.

[8] Taro Suzuki, Aart Middeldorp, and Tetsuo Ida. Level-confluence of conditional rewrite systems with extra variables in right-hand sides. In *Proceedings of the 6th International Conference on Rewriting Techniques and Applications*, volume 914 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 1995. doi:10.1007/3-540-59200-8_56.

[9] René Thiemann and Christian Sternagel. Certification of termination proofs using CeTA. In *Proceedings of the 22nd International Conference on Theorem Proving in Higher Order Logics*, volume 5674 of *Lecture Notes in Computer Science*, pages 452–468. Springer, 2009. doi:10.1007/978-3-642-03359-9_31.