Modular Church-Rosser Modulo

Jean-Pierre Jouannaud*

École Polytechnique
LIX, UMR CNRS 7161
91400 Palaiseau, France


Abstract. In [12], Toyama proved that the union of two confluent term-rewriting systems that share absolutely no function symbols or constants is likewise confluent, a property called modularity. The proof of this beautiful modularity result, technically based on slicing terms into an homogeneous cap and a so called alien, possibly heterogeneous substitution, was later substantially simplified in [5,11].

In this paper we present a further simplification of the proof of Toyama’s result for confluence, which shows that the crux of the problem lies in two different properties: a cleaning lemma, whose goal is to anticipate the application of collapsing reductions; a modularity property of ordered completion, that allows to pairwise match the caps and alien substitutions of two equivalent terms.

We then show that Toyama’s modularity result scales up to rewriting modulo equations in all considered cases.

1 Introduction

Let $R$ and $S$ be two rewrite systems over disjoint signatures. Our goal is to prove that confluence is a modular property of their disjoint union, that is that $R \cup S$ inherits the confluence properties of $R$ and $S$, a result known as Toyama’s theorem. In the case of rewriting modulo an equationnal theory also considered in this paper, confluence must be generalized as a Church-Rosser property. Toyama apparently anticipated this generalization by using the word Church-Rosser in his title.

A first contribution of this paper is a new comprehensive proof of Toyama’s theorem, obtained by reducing modularity of the confluence property to modularity of ordered completion, the latter being a simple property of disjoint unions. It is organized around the notion of stable equalizers, which are heterogeneous terms in which collapsing reductions have been anticipated with respect to the rewrite system $R^\infty \cup S^\infty$ obtained by (modular) ordered completion of $R \cup S$. Confluence of $R^\infty \cup S^\infty$ implies that equivalent terms have the same stable equalizers, made of a homogeneous cap which cannot collapse, and an alien stable substitution. This makes it possible to prove Toyama’s theorem by induction on the structure of stable equalizers.

A second contribution is a study of modularity of the Church-Rosser property when rewriting with a set of rules $R$ modulo a set of equations $E$. We prove that all rewrite relations introduced in the literature, class rewriting, plain rewriting modulo, rewriting modulo, normal rewriting and normalized rewriting enjoy a modular Church-Rosser

* Project LogiCal, Pôle Commun de Recherche en Informatique du Plateau de Saclay, CNRS, École Polytechnique, INRIA, Université Paris-Sud.

© Springer-Verlag Berlin Heidelberg 2006
property. We indeed show a more general generic result which covers all these cases. The proof is again obtained by applying selected results of the previous contribution to the rewrite system \( R \cup E^- \cup E^- \), obtained by orienting the equations in \( E \) both ways, which results in a confluent system when the original rewrite relation is confluent.

We introduce terms in Section 2 and recall the basic notions of caps and aliens in Section 3. The new proof of Toyama’s theorem is carried out in Section 4. Modularity of rewriting modulo is addressed in Section 5. Concluding remarks come in Section 6. We assume familiarity with the basic concepts and notations of term rewriting systems and refer to [1,11] for supplementary definitions and examples.

2 Preliminaries

Given a signature \( \mathcal{F} \) of function symbols, and a set \( \mathcal{X} \) of variables, \( T(\mathcal{F}, \mathcal{X}) \) denotes the set of terms built up from \( \mathcal{F} \) and \( \mathcal{X} \).

Terms are identified with finite labelled trees as usual. Positions are strings of positive integers, identifying the empty string \( \lambda \) with the root position. We use \( \mathcal{P} \text{os}(t) \) (resp. \( \mathcal{F} \mathcal{P} \text{os}(t) \)) to denote the set of positions (resp. non-variable positions) of \( t \), \( t(p) \) for the symbol at position \( p \) in \( t \), \( t'_p \) for the subterm of \( t \) at position \( p \), and \( t[u]_p \) for the result of replacing \( t'_p \) with \( u \) at position \( p \) in \( t \). We may sometimes omit the position \( p \), writing \( t[u] \) for simplicity. \( \mathcal{V} \text{ar}(t) \) is the set of variables occurring in \( t \).

Substitutions are sets of pairs \( (x,t) \) where \( x \) is a variable and \( t \) is a term. The domain of a substitution \( \sigma \) is the set \( \text{Dom}(\sigma) = \{ x \in \mathcal{X} \mid \sigma(x) \neq \lambda \} \). A substitution of finite domain \( \{x_1, \ldots, x_n\} \) is written as in \( \sigma = \{ x_1 \mapsto t_1, \ldots, x_n \mapsto t_n \} \). A substitution is ground if \( \sigma(x) \) is a ground term for all \( x \in \mathcal{X} \). We use greek letters for substitutions and \( \mathcal{F} \mathcal{P} \text{os}(t) \) at position \( \mathcal{V} \text{ar}(t) \) for the result of \( \sigma \).

Bijective substitutions are called variable renamings.

Given two terms \( s, t \), computing the substitution \( \sigma \) whenever it exists such that \( t = s\sigma \) is called matching, and \( s \) is then said to be more general than \( t \). This quasi-ordering is naturally extended to substitutions. Given to terms \( s, t \) their most general unifier whenever it exists is the most general substitution \( \sigma \) (unique up to variable renaming) such that \( s\sigma = t\sigma \).

A (plain) rewrite rule is a pair of terms, written \( l \rightarrow r \), such that \( l \not\in \mathcal{X} \) and \( \mathcal{V} \text{ar}(r) \subseteq \mathcal{V} \text{ar}(l) \). Plain rewriting uses plain pattern-matching for firing rules: a term \( t \) rewrites to a term \( u \) at position \( p \) with the rule \( l \rightarrow r \in R \) and the substitution \( \sigma \), written \( t \longrightarrow_{l \rightarrow r, \sigma}^* u \) if \( t|_p = l\sigma \) and \( u = t[r\sigma]_p \). A (plain) term rewriting system is a set of rewrite rules \( R = \{ l_i \rightarrow r_i \} \). An equation is a rule which can be used both ways. An equation \( x = s \) with \( x \in \mathcal{X} \) is collapsing. We use AC for associativity and commutativity, and \( \leftrightarrow_E \) for rewriting with a set \( E \) of equations.

The reflexive transitive closure of a relation \( \rightarrow \), denoted by \( \rightarrow^* \), is called derivation, while its symmetric, reflexive, transitive closure is denoted by \( \leftrightarrow^* \), or \( \leftrightarrow^*_R \) or \( \equiv_R \) when the relation is generated by a rewrite system \( R \). A term rewriting system \( R \) is confluent (resp. Church-Rosser) if \( t \rightarrow^* u \) and \( t \rightarrow^* v \) (resp. \( u \leftrightarrow^* v \)) implies \( u \rightarrow^* s \) and \( v \rightarrow^* s \) for some \( s \). The Church-Rosser property shall sometimes be used for some subset \( T \subset T(\mathcal{F}, \mathcal{X}) \), in which case \( u, v \) are assumed to belong to \( T \).