Title	Confluence Analysis for Term Rewriting via Commutation
Author(s)	新谷，喜楽
Citation	
Issue Date	2015-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/12660
Rights	
Description	Supervisor: 廣川 直，情報科学研究科，修士

Confluence Analysis for Term Rewriting via Commutation

Kiraku Shintani (1310032)

School of Information Science,
Japan Advanced Institute of Science and Technology

February 12, 2015

Keywords: term rewriting, confluence, automation.

Term rewriting is a simple Turing complete computational model, which underlies automated theorem proving (e.g. E, Vampire, Waldmeister) and declarative programming languages (CafeOBJ, Haskell, OCaml). Confluence is a fundamental property that ensures uniqueness of computational results, which plays a crucial role in applications. While in programming languages confluence guarantees well-definedness of functions, in theorem proving confluence is used for equational reasoning.

Rewriting and Confluecne. In this thesis we investigate automated confluence analysis for term rewriting. A term rewrite system (TRS) is a directed equational system on terms, computes a term to an another term. Not infrequently computation paths are different, the computational results are same. Here a natural question arises: Does such a uniqueness hold for any computation that starts from a term? Confluence addresses the issue.

Automated Confluence Proving. Research of confluence has a long history, and many powerful confluence criteria have been proposed [5, 7, 10, 11, 12, 13]. Especially a significant amount of research exists for the class of left-linear TRSs, which model much of functional programs. Left-linearity means that for every rewrite rules each variable in the left-hand side occurs exactly once. In 2009 the first automatic confluence tool ACP [3] appeared. This triggered a renewed interest in renovating confluence criteria in the aspect of computability and efficiency [1, 2, 3, 4, 6, 9, 14], and development of confluence tools (CSI [14] and Saigawa).

We are concerned with associativity and/or commutativity rules and the commutation property (see Figure 1) of rewrite systems.

Approach. In this thesis we propose a confluence analysis for left-linear TRSs via commutation. Commutation is a generalization of the confluence property (see Figure 1). The celebrated Commutation Theorem of Hindley [5] enables us to decompose the confluence problem of a complex TRS into a group of commutation problems of its subsystems. As direct methods for commutation, we employ confluence criteria including rule labeling [1] and the Church-Rosser modulo theorem [8], recasting them in commutation criteria. In

Figure 1: Confluence (left) and commutation (right)

order to derive the power of the Church-Rosser modulo theorem we have to perform equational unification, automation of which is one of the highlights of this thesis. In addition to those core contributions, we introduce several techniques useful for improving power and efficiency of confluence analysis. We remark that left-linearity of TRSs is an essential property of commutation, in fact many commutation criteria require left-linearity.

Contributions. Here we list the main contributions of the thesis:

- a confluence proof by Church-Rosser modulo associativity and/or commutativity theories (Chapter 3),
- a commutation-based confluence analysis (Chapter 4),
- composability decomposition (Chapter 4),
- redundant rule elimination (Chapter 5),
- signature extension for commutation (Chapter 6), and
- the powerful confluence tool CoLL:

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

References

[1] T. Aoto. Automated confluence proof by decreasing diagrams based on rule-labelling. In *Proc. 21st RTA*, volume 6 of *LNCS*, pages 7–16, 2010.

[2] T. Aoto and Y. Toyama. A reduction-preserving completion for proving confluence of non-terminating term rewriting systems. *LMCS*, 8(1):1–29, 2012.

[3] T. Aoto, J. Yoshida, and Y. Toyama. Proving confluence of term rewriting systems automatically. In *RTA 2009*, volume 5595 of *LNCS*, pages 93–102, 2009.

[4] B. Felgenhauer. Deciding confluence of ground term rewrite systems in cubic time. In *Proc. 23rd RTA*, LIPIcs, pages 165–175, 2012.

[5] J. R. Hindley. *The Church-Rosser Property and a Result in Combinatory Logic*. PhD thesis, University of Newcastle-upon-Tyne, 1964.

[6] N. Hirokawa and A. Middeldorp. Decreasing diagrams and relative termination. *Journal of Automated Reasoning*, 47(4):481–501, 2011.

[7] G. Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *Journal of the ACM*, 27(4):797–821, 1980.

[8] J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal on Computing*, 15(4):1155–1194, 1986.

[9] D. Klein and N. Hirokawa. Confluence of non-left-linear TRSs via relative termination. In *Proc. 18th LPAR*, volume 7180 of *LNCS*, pages 258–273, 2012.

[10] D.E. Knuth and P.B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.

[11] Y. Toyama. Commutativity of term rewriting systems. In K. Fuchi and L. Kott, editors, *Programming of Future Generation Computers II*, pages 393–407. North-Holland, 1988.

[12] V. van Oostrom. Confluence by decreasing diagrams. *Theoretical Computer Science*, 126(2):259–280, 1994.

[13] V. van Oostrom. Developing developments. *Theoretical Computer Science*, 175(1):159–181, 1997.

[14] H. Zankl, B. Felgenhauer, and A. Middeldorp. CSI — a confluence tool. In *Proc. 23rd CADE*, pages 499–505. Springer, 2011.