A CRITICAL PAIR CRITERION FOR COMPLETION MODULO A CONGRUENCE[1]

Leo Bachmair
Department of Computer Science
SUNY at Stony Brook
Stony Brook, NY 11794, U.S.A.

Nachum Dershowitz
Department of Computer Science
University of Illinois
Urbana, IL 61801, U.S.A.

Extended Abstract

Rewrite systems are collections of directed equations (rules) used to compute by repeatedly replacing subterms in a given formula until a simplest form possible (normal form) is obtained. Many formula manipulation systems such as REDUCE or MACSYMA use equations for simplification in this manner. Canonical (i.e., terminating Church-Rosser) rewrite systems have the property that all equal terms (and only equal terms) simplify to an identical normal form. Deciding validity in theories for which canonical systems are known is thus easy and reasonably efficient. A number of canonical systems have been derived with the Knuth-Bendix completion procedure [5]. Unfortunately, the Knuth-Bendix procedure can not be applied to axioms such as commutativity that induce non-terminating rewrite sequences. There are also some practical limitations in its handling of associativity, as pointed out by Peterson and Stickel [6]. Associativity and commutativity are typical equations that are more naturally viewed as "structural" axioms (defining a congruence relation on terms) rather than as "simplifiers" (defining a reduction relation).

Given a set of axioms A and a rewrite system R, we denote by R_A the corresponding relation of *rewriting modulo* A, defined as the application of rules in R via A-matching. For example, if A consists of the associativity and commutativity axioms for addition, and the rewrite system R consists of a single rule $f(x,x) \to x$, then $f(x+y,y+x)$ can not be rewritten by R, whereas it can be rewritten to $x+y$ (and $y+x$) by R_A. Extensions of the Knuth-Bendix procedure to rewriting modulo a congruence have been described by Peterson and Stickel [6] (for sets A of associativity and commutativity axioms), by Jouannaud and Kirchner [3], and Bachmair and Dershowitz [1]. The fundamental operations in these procedures are A-matching and A-unification.

Two terms s and t are said to be *A-unifiable* if and only if there is a substitution σ (called an *A-unifier*), such that $s\sigma$ and $t\sigma$ are equivalent with respect to A. The above-mentioned completion procedures apply to theories for which complete sets of A-unifiers can be computed. If $u \to v$ and $s \to t$ are rules and p is a non-variable position in u, such that u/p and s are A-unifiable with a complete set of unifiers Σ, then the "rewriting ambiguity" $v\sigma \leftarrow_R u\sigma \rightarrow_{R_A} u\sigma[t\sigma]$ determines an *A-critical pair* $v\sigma = u\sigma[t\sigma]$, for each σ in Σ.

Completion augments a given rewrite system R by so-called "extended" rules[2] and then systematically computes A-critical pairs to check whether the two terms $v\sigma$ and $u\sigma[t\sigma]$ reduce to an identical normal form. If the test is successful for all critical pairs, then the rewrite system is canonical (in the sense of defining normal forms that are unique up to equivalence in A). Otherwise, the offending equations have to be turned into rules and critical pair computation continues with the new rules. (It is possible that endless new rules are generated and completion does not terminate. Completion may even fail, when an equation can not be oriented into a rule.) The most expensive part of completion is the reduction of terms to normal form. Critical pairs in which both terms reduce to identical normal forms are redundant. Various techniques, called *critical pair criteria*, have been proposed for standard completion for detecting redundancies more efficiently than by normalization of terms (see Bachmair and Dershowitz [2] for an overview). We sketch a similar technique for rewriting modulo a congruence.

[1]This research was supported in part by the National Science Foundation under grant DCR 85-13417.

[2]In the case of associative-commutative completion, extended rules are of the form $f(f(s,t),x) \to f(u,x)$, where x is a new variable and $f(s,t) \to u$ is a (non-extended) rule for which f is an associative-commutative operator.

A rewrite step $s \rightarrow_R t$ is said to be *blocked* (with respect to A) if it is by application of some rule with a substitution σ, such that no term $x\sigma$ can be rewritten by R_A. For example, if R contains rules $x + 0 \rightarrow 0$ and $-(x + y) \rightarrow -x + -y$, then the rewrite step $-((x + 0) + y) \rightarrow_R -(x + 0) + -y$ is not blocked, because the term $x + 0$, which is substituted for x, can be rewritten by R_A. Non-blocked rewrite steps can be replaced by a sequence of *blocked* steps. For instance, the above rewrite step can be replaced by $-((x + 0) + y) \rightarrow_{R_A} -(x + y) \rightarrow_{R_A} -x + -y \leftarrow_{R_A} -(x + 0) + -y$.

We say that an A-critical pair is blocked if both rewrite steps in the corresponding rewriting ambiguity $v\sigma \leftarrow_R u\sigma \rightarrow_{R_A} u\sigma[t\sigma]$ are blocked. All non-blocked A-critical pairs that are obtained from non-extended rules are redundant and can be disregarded by completion. The restriction to non-extended rules is crucial, though in the case of associative-commutative completion blocking can also be applied in a restricted form to extended rules. Specifically, A-critical pairs obtained by applying an extended rule $f(s, x) \rightarrow f(t, x)$ with a substitution σ, such that $y\sigma$ can be rewritten by R_A, for some variable y in s, can also be disregarded. It may be necessary, on the other hand, to instantiate the "extension variable" x by a term that can be rewritten.

For example, consider the set of two rules $a+b \rightarrow c$ and $(a+a)+(b+b) \rightarrow d$. The only A-critical pairs are those involving the extended rules $(a+b)+x \rightarrow c+x$ and $((a+a)+(b+b))+x \rightarrow d+x$, and require that the extension variable x be instantiated by a term that can be rewritten. But even though all A-critical pairs are non-blocked, the system is not canonical, as the term $(a+b)+(a+b)$ has two different normal-forms, $c + c$ and d, that are not equivalent with respect to associativity and commutativity.

Experiments that we have run using the associative-commutative completion procedure implemented in the rewrite rule laboratory RRL [4] indicate the usefulness of blocking. Results are summarized in Table 1 (the last column of the table refers to the ratio b/t of the respective times needed to obtain a canonical system with and without blocking).

Table 1. Associative-Commutative Completion

	STANDARD	BLOCKING			
	Critical pairs	Critical pairs	Blocked	Redundant	b/t
Abelian groups	86	95	50	45	0.75
Associative-commutative rings	248	317	151	166	0.83
Modules	416	462	271	191	0.68
Lattices	151	151	63	88	0.38
Boolean rings	99	109	38	71	0.50
Non-deterministic machines	284	284	131	153	0.45

References

[1] L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. To appear in *Theor. Comput. Sci.* (1988).

[2] L. Bachmair and N. Dershowitz. Critical pair criteria for completion. *J. Sym. Comput.* (1988) **6**:1-18.

[3] J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM J. Comput.* (1986) **15**:1155–1194.

[4] D. Kapur, D.R. Musser, and P. Narendran. Only prime superpositions need be considered in the Knuth-Bendix completion procedure. *J. Sym. Comput.* (1988) **6**:19-36.

[5] D. Knuth and P. Bendix. *Simple word problems in universal algebras.* In *Computational Problems in Abstract Algebra*, ed. J. Leech, Pergamon Press, Oxford, 1970, pp. 263–297.

[6] G. Peterson and M. Stickel. Complete sets of reductions for some equational theories. *J. ACM* (1981) **28**:233–264.