Normalised Rewriting and Normalised Completion

Claude Marché
LRI, URA 410 du CNRS & INRIA*
INRIA Rocquencourt — Bât. 11
BP 105
78153 Le Chesnay cedex
France

Abstract

We introduce normalised rewriting, *a new rewrite relation. It generalises former notions of rewriting modulo E, dropping some conditions on E. For example, E can now be the theory of identity, idempotency, the theory of Abelian groups, the theory of commutative rings. We give a new completion algorithm for normalised rewriting. It contains as an instance the usual AC completion algorithm, but also the well-known Buchberger's algorithm for computing standard bases of polynomial ideals.*

We investigate the particular case of completion of ground equations, In this case we prove by a uniform method that completion modulo E terminates, for some interesting E. As a consequence, we obtain the decidability of the word problem for some classes of equational theories. We give implementation results which shows the efficiency of normalised completion with respect to completion modulo AC.

1 Introduction

Equational axioms are very common in most sciences, including computer science. Equations can be used for reasoning, by using Leibniz's law of replacing equals by equals. An *equational proof* from s to t may therefore use the equations both ways. In contrast, *rewrite proofs* restrict their use to be one way, by rewriting according to a well-founded ordering on terms from both s and t. This strategy amounts to orient the equations into rewrite rules via the ordering. Transforming an equational proof into a rewrite proof needs replacing the undesirable patterns such as $s \leftarrow u \rightarrow t$ by appropriate rewrite proofs. To achieve this purpose, the key step is to compute the so-called critical pairs by overlapping left-hand sides of rules, then rewriting the obtained term via each one of the two rules. When such critical pairs do not enjoy a rewrite proof, they may be simplified, then oriented and added as new rules. Rules may be simplified as well in order to obtain a reduced set. This process is called Knuth-Bendix completion [21]. In completion, the axioms used are therefore in a constant state of flux; these changes are usually expressed as inference rules, which add a dynamic character to establishing the existence of rewrite proofs.

A basic assumption of this technique is that rewriting terminates for every input term. When the set of equations contains the associativity and commutativity axioms (hereafter denoted by AC), this assumption cannot be fulfilled. Lankford and Ballantyne [23], and Peterson and Stickel [32] have shown how to resolve this difficulty by building associativity and commutativity in the AC-rewriting process (via AC pattern matching), as well as in the computation of AC critical pairs (via AC-unification). This has been further generalised to an arbitrary theory split into an E-part and an S-part provided S-unification is finitary and the sub-term ordering modulo S is noetherian [3, 14].

This technique excludes therefore some important sets of axioms like the identity law ($x+0=x$, denoted AC1), group theory, idempotency ($x+x=x$, denoted ACI), etc, to be part of the set S. Indeed, in all these cases, S-rewriting does not terminate in general. For example, rewriting modulo AC1 yields the following infinite derivation, using the rule $-(x+y) \rightarrow (-x)+(-y)$ for computing the inverse of a sum:

$$\begin{array}{rll} -0 & =_{\mathrm{AC1}} -(0+0) & \rightarrow (-0)+(-0) \\ & =_{\mathrm{AC1}} -(0+0)+(-0) & \text{etc...} \end{array}$$

It is possible to overcome this difficulty using *constrained* rewriting [6, 15, 20]. Unfortunately, this ap-

*This work is partly supported by the "GDR Programmation du CNRS", the ESPRIT Working Group "Compass" and the ESPRIT Basic Research Action "TYPES". E-mail: `Claude.Marche@inria.fr`

proach does nor work for other theories mentioned above: for any rule $l \to r$, taking ACI for S yields:

$$\begin{array}{lll} l & =_{\mathrm{ACI}} l+l & \to l+r \\ & =_{\mathrm{ACI}} l+l+r & \text{etc}\ldots \end{array}$$

and taking now AG (Abelian groups theory) for S yields:

$$\begin{array}{lll} 0 & =_{\mathrm{AG}} l+(-l) & \to r+(-l) \\ & =_{\mathrm{AG}} l+(-l)+r+(-l) & \text{etc}\ldots \end{array}$$

hence in both cases, rewriting on congruence classes *never* terminates.

In this paper, we present a new rewriting technique, called *normalised* rewriting, which assumes that the theory E is presented by an AC-convergent set of rules. The main idea is: before rewriting a term, one has to reduce it to its normal form for E. We show that this rewrite relation generalises the AC1-constrained rewriting, but also allows to rewrite modulo idempotency, groups theory, and other

This paper is organised as follows: in section 2 we define what is normalised rewriting, and we show that the termination of this new rewrite relation can be checked by a reduction ordering compatible with AC, and only with AC not with the whole theory E, for which such an ordering does not exist in general. We give in section 3 a general scheme of a completion procedure, which can be further instantiated, first into a completion procedure modulo an arbitrary E, and second into completion procedures modulo some particular E, like groups or commutative rings, which are more efficient that the general one. We prove in section 4 that our particular completion procedures modulo some theories always terminate when the input set of equations is *ground* (*i.e.* has no variables). We obtain an alternative proof of the decidability of the word problem in finitely presented Abelian groups and finitely presented commutative rings, but our results are in fact more general since we do not need that the generators are only constants, as it is the case in finitely presented groups or rings. Such a result extended to non-constant generators has already been proved for ground theories modulo AC (for an arbitrary number of AC operators) [25, 29]. Finally, we give in section 5 some example of normalised completion and we show in particular some interesting benchmarks and also how normalised completion can be used to compute standard bases of polynomial ideals.

We can summarise the main results of this paper:

- given an equational theory S which possesses a convergent system, there is a completion algorithm which, given a set of equations E, computes a set of rules R such that S-normalised rewriting by R decides the equality modulo $E \cup S$: for all s and t, $s =_{E\cup S} t$ if and only if

$$s \xrightarrow[R/S]{*} u \equiv v \xleftarrow[R/S]{*} t$$

- an equational theory presented by $C \cup E$ where C is a set of ground equations and E is either AC, AC1, ACI, AC1I, AG, CR, BR or FF(p) has a decidable word problem, more precisely possesses an E-normalised rewriting system.

We have omitted most of the proofs in this paper, all the results are proved completely in the author's PhD thesis [27].

2 Normalised rewriting

In this section we introduce the new notion of normalised rewriting. We recall first the usual notions on rewriting, in particular modulo AC.

2.1 Basic definitions

We first recall briefly the basic definitions on rewriting. Our notations and definitions are consistent with those given in the survey of Dershowitz and Jouannaud [12].

We denote $\mathcal{T}(\mathcal{F},\mathcal{X})$, or $\mathcal{T}$ for short, the set of terms over a signature $\mathcal{F}$ and variables $\mathcal{X}$. We denote $\mathcal{P}\mathrm{os}(s)$ and $\mathcal{FP}\mathrm{os}(s)$ respectively the set of positions and non variable positions of a term s. We denote Λ the top position. The sub-term of a term s at position p is denoted by $s|_p$, and $s[t]_p$ is the term obtained by putting t at position p in s. We denote substitutions by Greek letters, $s\sigma$ is the application of σ on s.

An *equation* is a pair of terms, denoted $s = t$. An equation is *valid* in an $\mathcal{F}$-algebra A if for any $\mathcal{F}$-morphism $g : \mathcal{T} \to A$ we have $g(s) = g(t)$. An equation $s = t$ is a consequence of a set of equations E if $s = t$ is valid in every algebra that validates E. The set of consequences of E, denoted $\mathcal{T}h(E)$ is the *equational theory* of E.

The *equality modulo* E, generated by a set of equations E, is the smallest congruence containing E, denoted $=_E$. Because of Birkhoff's theorem [8]: $s = t$ is a consequence of E if and only if $s =_E t$, we may usually confuse E, $\mathcal{T}h(E)$ and $=_E$.

An important example is the associative-commutative theory, denoted by AC. Over a signature $\mathcal{F}$ which contains a subset $\mathcal{F}_{\mathrm{AC}}$ of binary symbols, AC is the set $\{f(x,y) = f(y,x), f(f(x,y),z) = f(x,f(y,z)) \mid f \in \mathcal{F}_{\mathrm{AC}}\}$. Usually AC operators are used in infix notation ($+$, $*$, etc.).

Congruences classes modulo AC can be represented as *flat* terms. This representation is usually preferred in implementations, and is also a useful representation from a theoretical point of view, for example in AC unification algorithms. In this article, we will consider that terms are flattened with respect to the AC symbols of the signature. Formal and complete definitions of flattening and rewriting on flat terms can be found in [13, 17, 27]. Two terms are equal modulo AC if and only if their flat forms are equivalent modulo the *permutation congruence* (denoted $\equiv$), that is the equivalence modulo permutation of direct sub-terms of AC symbols.

We say that two terms s and t are unifiable modulo a theory E if there exists a substitution σ such that $s\sigma =_E t\sigma$. Main results on E-unification may be found in the survey edited by Kirchner [19]. We denote $\mathrm{CSU}_E(s,t)$ a complete set of E-unifiers of s and t.

We use rewriting on flat terms, that is we say that s *rewrites* to t by $l \rightarrow r$ at position $p \in \mathcal{FP}\mathrm{os}(s)$ (denoted as $s \xrightarrow[l\rightarrow r]{p} t$) if there exists a substitution σ such that $s|_p = l\sigma$ and $t = s[r\sigma]_p$, or $s|_p = (l+x)\sigma$ and $t = s[(r+x)\sigma]_p$ if $\mathcal{H}\mathrm{ead}(l) = + \in \mathcal{F}_{\mathrm{AC}}$ and $x \notin \mathcal{V}\mathrm{ar}(l)$. This way of defining the rewrite relation builds in the use of extended rules "à la Peterson-Stickel" [32]: indeed, when using flat rewriting, we do not need to introduce *extended* rules, it greatly simplifies the proof of completeness of completion. Moreover, not adding extension rules prevents introduction of new variables, which is essential when completing a set of ground equations. However we have to generalise the notion of overlapping: two rules which have the same AC symbol $+$ at the top overlap if they overlap in the standard way or if their extensions overlap. For example, there is a critical pair between $a+b \rightarrow d$ and $a+c \rightarrow e$ since $a+b+c$ can be rewritten either to $d+e$ or $b+e$. We still denote by CP_E the set of critical pairs modulo E corresponding to this generalised notion of overlapping (assuming that E contains at least AC).

A set of rules R is said to be convergent (modulo AC) if $\xrightarrow[R]{}$ is noetherian and confluent.

2.2 Definition of normalised rewriting

Let us assume now that the theory E is given by a convergent set of rules S (modulo AC).

Definition 2.1 *Let S be an AC-convergent rewrite system. Let us denote by $s\downarrow_S$ the S-normal form of a term s. The S-*normalised rewrite relation*, denoted as $s \xrightarrow[l\rightarrow r/S]{p} t$, is defined by* $\begin{cases} s' = s\downarrow_S \\ s' \xrightarrow[l\rightarrow r]{p} t \end{cases}$.

Example 2.2 *Assume $S = \mathrm{AC1}(+,0) = \{x+0 \rightarrow x\}$. Assume we would like to rewrite by $R = \{-(x+y) \rightarrow (-x)+(-y)\}$. We have $-(a+b) \xrightarrow[R/S]{} (-a)+(-b)$ but we can not rewrite $-(0+b)$ to $(-0)+(-b)$ because the S-normal form of $-(0+b)$ is $(-b)$ which is not an instance of $-(x+y)$.*

We see on this example that the idea of normalised rewriting captures the notion of AC1-constrained rewriting [15].

Example 2.3 *Assume S is the convergent rewrite system of commutative rings theory, that is*

$$\begin{cases} x+0 \rightarrow x & x+(-x) \rightarrow 0 \\ -0 \rightarrow 0 & -(-x) \rightarrow x \\ -(x+y) \rightarrow (-x)+(-y) & x*1 \rightarrow x \\ (x+y)*z \rightarrow (x*y)+(y*z) & x*0 \rightarrow 0 \\ x*(-y) \rightarrow -(x*y) & \end{cases}$$

Assume we have $R = \{X*X \rightarrow Y\}$ where X and Y are some constants. Then $X*X*X \xrightarrow[R/S]{} X*Y$ but $X*(X-Y)+(-X*X)$ can not be rewritten since the S-normal form of $X*(X-Y)+(-X*X)$ is $-X*Y$ and is not reducible by R.

We see in this case that normalised rewriting captures the notion of polynomial reduction used in standard basis computation, where the distributivity law is applied before the rules.

2.3 Termination of normalised rewriting

Proving termination of (usual) rewriting modulo E requires an ordering compatible with E. Such an ordering does not exist in general. For example there is no reduction ordering compatible with idempotency as shown in the introduction.

One interesting property of our new definition of rewriting is that we only need a reduction ordering compatible with AC. Such an ordering can be defined in various ways. For general notions on orderings and termination, we refer to [11]. For definitions of AC-compatible orderings, see [5, 7, 10, 29, 30].

From now, we assume given an AC reduction ordering $\succeq$, and a set of rules S, convergent modulo AC, such that $\rightarrow_S \subseteq \succ$ (that is the termination of S modulo AC can be proved by $\succeq$). The following proposition is straightforward:

Proposition 2.4 *Let R be a set of rules such that for all $l \rightarrow r$ in R, $l \succ r$. Then the S-normalised rewrite relation $\xrightarrow[R/S]{}$ is noetherian.*

In section 4, when trying to complete a set of *ground* equations, we need to prevent failure cases. For, we assume that our AC ordering we use is *total on ground terms*. It is not very easy to define such an ordering, but it is possible [27, 29, 30].

3 Normalised completion

We give in this section a set of inference rules for completing a set of equations into a normalised rewrite system. The completeness is proved by the now customary normalisation proof method [3, 4]. We first need to introduce the notion of *normalising pairs*, which in some sense will replace the usual notion of orientation in completion procedures.

3.1 Normalising pairs

We assume given a convergent set of rules S, together with a reduction ordering $\succ$ such that $\xrightarrow[S]{} \subseteq \succ$. We have now to be given a *proof reduction ordering* on the *algebra of proofs*.

Definition 3.1 *The algebra of equational proofs is generated by the elementary proofs $s \xleftrightarrow[\equiv]{} t$ (AC step), $s \xleftrightarrow[l=r]{\sigma,p} t$ (equational step), $s \xrightarrow[l\rightarrow r]{\sigma,p} t$ (rewrite step), $s \xrightarrow[S]{} t$ (S-normalising step); and the concatenation of proofs, denoted $P.Q$, where the last term of P is assumed to equal the first of Q. We say that a proof is* in $E \cup R$ *if its equations (resp. its rules) are in E (resp. in R).*

An ordering $\succ_{\mathcal{P}}$ on the algebra of equational proofs is a proof reduction ordering *if it satisfies:*

- *monotonicity property: if $P \succ_{\mathcal{P}} P'$ then $Q.P.R \succ_{\mathcal{P}} Q.P'.R$;*
- *it is noetherian.*

We assume now given a proof ordering $\succ_{\mathcal{P}}$. The conditions below are exactly the ones we need to prove next the completeness theorem.

Definition 3.2 *A function that maps a pair of terms (u, v) to a pair $(\Theta(u, v), \Psi(u, v))$ where $\Theta(u, v)$ is a set of equations and $\Psi(u, v)$ a set of rules, is called an S-normalising pair (w.r.t. $\succ_{\mathcal{P}}$) if*

- *for any elementary S-irreducible equational proof of the form $s \xleftrightarrow[u=v]{} t$ or $s \xrightarrow[u\rightarrow v]{} t$, there exists a smaller proof (w.r.t. $\succ_{\mathcal{P}}$) in $\Theta(u, v) \cup \Psi(u, v)$ between s and t;*
- *for all $l \rightarrow r \in \Psi(u, v)$, $\Theta(l, r) \subseteq \Theta(u, v)$ and $\Psi(l, r) \subseteq \Psi(u, v)$.*

3.2 Inference rules for normalised completion

As now customary, we describe the completion process by a set of inference rules (Figure 1). $\succ$ is an AC reduction ordering, E is a set of equations and R is a set of rules. The rule DEDUCE computes critical pairs modulo some equational theory T that we can choose arbitrarily between AC and S. This is a very important point for two reasons:

- S may not be *finitary* with respect to unification. For example, S may contain distributivity law, and we know that unification modulo ACD is undecidable [19]. In such a case we may use $T =$ AC.
- It is known that AC1-unification and AC1I-unification lead to complete sets of unifiers which are usually much smaller than AC unification [19]. AC1I-unification is even more efficient than AC1-unification since it is a *unitary* theory.

Definition 3.3 *An* S-normalised completion algorithm *is an algorithm which takes as input a set of equations E_0 and an AC reduction ordering $\succ$ and produces a (finite or infinite) sequence $(E_n; R_n)$ where $R_0 = \emptyset$ and for all i, $E_i; R_i \vdash E_{i+1}; R_{i+1}$. Let:*

$$E_\infty = \bigcup_{n=0}^{\infty} \left(\bigcap_{i=n}^{\infty} E_i \right) \; , \; R_\infty = \bigcup_{n=0}^{\infty} \left(\bigcap_{i=n}^{\infty} R_i \right)$$

E_∞ and R_∞ are respectively the sets of persisting equations and the set of persisting rules. We say that the algorithm fails *if E_∞ is not empty and* succeeds *otherwise, it* diverges *if the sequence is infinite.*

3.3 Fairness and Completeness

Fairness is fundamental in completion procedures, it expresses completeness of the search strategy.

Orient	$E \cup \{u = v\}; R \ \vdash \ E \cup \Theta(u,v); R \cup \Psi(u,v)$	if $u = u{\downarrow_S},\ v = v{\downarrow_S},\ u \succ v$
Deduce	$E; R \ \vdash \ E \cup \{u = v\}; R$	if $u = v \in CP_T(R)$
Normalize	$E \cup \{u = v\}; R \ \vdash \ E \cup \{u{\downarrow_S} = v{\downarrow_S}\}; R$	
Delete	$E \cup \{u = v\}; R \ \vdash \ E; R$	if $u =_{\mathrm{AC}} v$
Compose	$E; R \cup \{u \to v\} \ \vdash \ E; R \cup \{u \to v'\}$	if $v \xrightarrow[R/S]{} v'$
Simplify	$E \cup \{u = v\}; R \ \vdash \ E \cup \{u' = v\}; R$	if $u \xrightarrow[R/S]{} u'$
Collapse	$E; R \cup \{u \to v\} \ \vdash \ E \cup \{u' = v\}; R$	if $l \to r \in R,\ u \xrightarrow[l \to r/S]{\theta,p} u'$,
	$p \neq \Lambda$ or $p = \Lambda$ and θ is not a renaming or $p = \Lambda,\ \theta$ renaming and $u \succ r\theta$	

Figure 1: Inference rules of normalised completion

Definition 3.4 *A derivation $E_0; R_0 \ \vdash \ E_1; R_1 \ \vdash \ \ldots$ is* fair *if all persisting critical pairs are computed, i.e.*

$$CP_T(R_\infty) \subseteq \bigcup_{i=0}^{\infty} E_i$$

A completion algorithm is fair if all sequences that it produces are fair.

In practice, it is worth to use the simplification rules as much as possible. This yields sets of rules which are inter-reduced, an important property as far as the uniqueness of the completion result is concerned.

The set of inference rules induces a set of reduction rules on proofs, it is omitted here because of space limitation. We assume now that we have a proof reduction ordering $\succ_{\mathcal{P}}$ such that these rules decrease with respect to $\succ_{\mathcal{P}}$.

Theorem 3.5 *Assume we have an S-normalising pair (Θ, Ψ) (w.r.t. $\succ_{\mathcal{P}}$). Assume that the completion is fair and succeeds. Then for all s and t, $s =_{E_0 \cup \mathrm{AC} \cup S} t$ if and only if*

$$s \xrightarrow[R_\infty/S]{*} u \equiv v \xleftarrow[R_\infty/S]{*} t$$

This result is proved by the proof normalisation method [2–4, 27].

3.4 A general S-normalising pair

We show in this subsection how one can define an S-normalising pair for an arbitrary S.

Let $s{\downarrow_p}$ be the result of S-normalising s at position p, that is $s[(s|_p){\downarrow_S}]_p$, and $c(s,p,t)$ be the multi-set $\{s\}$ if $s{\downarrow_p} = s$ and $\{s{\downarrow_p}, t{\downarrow_p}\}$ otherwise.

We use the following proof ordering: the complexity of a proof is the multi-set of the complexities of its elementary subproofs, defined by

$$\begin{array}{rcl} C(s \xleftrightarrow[\equiv]{} t) & = & \langle \{\bot, \{s\}, \bot, \bot\rangle \\ C(s \xleftrightarrow[l=r]{\sigma,p} t) & = & \langle \{s{\downarrow_p}, t{\downarrow_p}\}, \{s,t\}, \bot, \bot\rangle \\ C(s \xrightarrow[l \to r]{\sigma} t) & = & \langle \{c(s,p,t), \{s\}, l, r\sigma\rangle \\ C(s \xrightarrow[S]{} t) & = & \langle \{\bot\{s\}, \bot, \bot\rangle \end{array}$$

where $\bot$ is a new minimal element, two elementary complexities are compared in the lexicographic extension of the orderings $\succ_{mul}$ for the first and second components, encompassment for the third, $\succ$ for the fourth. Proofs are compared in the multi-set extension of the above ordering. This ordering is noetherian, since it is built up from noetherian orderings with the functionals *lex* and *mul* which preserve well foundedness.

Definition 3.6 *Let $\Theta_{\mathrm{gen}}(u,v)$ be the set of equations $u\theta[r\theta]_q = v\theta$ where $q \in \mathcal{FP}\mathrm{os}(u)$, $l \to r \in S$, $\theta \in CSU_{\mathrm{AC}}(u|_q, l)$ S-irreducible, and the equations*

theory S	convergent system	$\Theta_S(u,v)$
$AC1(+,0)$	$x+0\to x$	$\{u\theta=v\theta \mid \theta=x\mapsto 0, u\trianglerighteq x+w\}$
$ACI(+)$	$x+x\to x$	$\{u\sigma=v\sigma \mid \sigma\in \mathrm{CSU}_{\mathrm{AC}}(l_1,l_2), u\trianglerighteq l1+l2\}$
$AC1I(+,0)$	$x+x\to x,\ x+0\to x$	$\Theta_{AC1}(u,v)\cup\Theta_{ACI}(u,v)$
$AC0(.,0)$	$x.0\to 0$	$\{u\theta=v\theta \mid \theta=x\mapsto 0, u\trianglerighteq x.w\}$
$ACN(+,0)$	$x+x\to 0$	$\{u\sigma=v\sigma \mid \sigma\in \mathrm{CSU}_{\mathrm{AC}}(l_1,l_2), u\trianglerighteq l1+l2\}$

Figure 2: Set Θ_S for some simple theories

$l\theta[v\theta]_q = r\theta$ *where* $q\in\mathcal{FP}\mathrm{os}(l)$, $q\neq\Lambda$, $l\to r\in S$, $\theta\in CSU_{\mathrm{AC}}(u,l|_q)$ *S-irreducible; and* $\Psi_{\mathrm{gen}}(u,v)$ *be* $\{u\to v\}$.

Proposition 3.7 *The pair* $(\Theta_{\mathrm{gen}},\Psi_{\mathrm{gen}})$ *is S-normalising with respect to the proof ordering defined above, for any AC-convergent set of rules S.*

Example 3.8 *Assume* $S=\{z+0\to z\}$. *Let us compute* $\Theta_{\mathrm{gen}}(-(x+y),(-x)+(-y))$*: we have to unify modulo AC the terms* $x+y$ *and* $z+0$. *This leads to 4 most general unifiers:*

$$\left\{\begin{array}{l}x\mapsto v_1\\ y\mapsto 0\\ z\mapsto v_1\end{array}\right.\left\{\begin{array}{l}x\mapsto 0\\ y\mapsto v_1\\ z\mapsto v_1\end{array}\right.\left\{\begin{array}{l}x\mapsto v_1\\ y\mapsto v_2+0\\ z\mapsto v_1+v_2\end{array}\right.\left\{\begin{array}{l}x\mapsto v_1+0\\ y\mapsto v_2\\ z\mapsto v_1+v_2\end{array}\right.$$

The last two are S-reducible so we ignore them. Hence $\Theta_{\mathrm{gen}}(-(x+y),(-x)+(-y))$ *contains only the equations* $-x=(-x)+(-0)$ *and* $-y=(-0)+(-y)$.

One can remark that we obtain a set which is the same as the set of *forbidden instances* in AC1-constrained completion [15].

3.5 Optimised normalising pairs for some simple theories

When using normalised rewriting modulo a fixed S, we can optimise the definition of the general normalising pair. In particular if the rules of S are left-linear, we can avoid the use of AC unification. Figure 2 shows definitions of Θ_S where S is either AC1, ACI, AC1I, AC0 or ACN, and in all these cases $\Psi_S(u,v)=\{u\to v\}$.

Proposition 3.9 *If S is either AC1, ACI, AC1I, AC0 or ACN, the above defined mappings* Θ_S, Ψ_S *are S-normalising.*

3.6 Optimised normalising pair for commutative groups and rings by symmetrisation

When S contains at least Abelian groups theory, we can optimise much further the normalising pair by using *symmetrisation*. The idea is that in an equation $u_1+\cdots+u_n=v_1+\cdots+v_m$, we may move one term from one side to the other changing its sign. This notion is inspired by [24]. We use the abbreviation nt for $\underbrace{t+\cdots+t}_{n\text{ times}}$.

Definition 3.10 *The* symmetrisation *of a pair* (u,v) *is obtained in the following way: if* $u=n_1u_1+\cdots+n_ku_k$ *and* $v=m_1t_1+\cdots+m_lt_l$, *with* $\forall j\geq 2 u_1\succ u_j$ *and* $\forall j u_1\succ t_j$, *then* $\mathrm{sym}(u,v)=(n_1,u_1,-n_2u_2-\cdots-n_ku_k+m_1t_1+\cdots+m_lt_l)$. *If there is no maximum* u_i *or* v_j, $\mathrm{sym}(u,v)$ *is undefined.*

Definition 3.11 *For a pair* (u,v) *that has a symmetrisation* (n,s,t), *let* $\Psi_{AG}(u,v)=\{ns\to t, -s\to(n-1)s+(-t)$ *if* $n\geq 2\}$ *and* $\Theta_{AG}(u,v)=\Theta_{AC1}(ns,t)\cup\Sigma_1(ns,t)\cup\Sigma_2(ns,t)$ *where* $\Sigma_1(u,v)=CP_{\mathrm{AC}}(u\to v, x+(-x)\to 0)$ *and* $\Sigma_2(u,v)=\{u\sigma=v\sigma \mid \sigma=x\mapsto 0$ *or* $-y$ *or* $y+z$ *if* $u\trianglerighteq -x\}$. *If* (u,v) *does not have a symmetrisation, the equation* $u=v$ *will be considered as not orientable.*

Proposition 3.12 *If the term ordering is an RPO with a precedence* $->+>0$ *and all other symbols greater than* $-$, *then the pair* (Θ_{AG},Ψ_{AG}) *defined above is AG-normalising.*

This symmetrisation technique improves a lot over standard AC completion when the set of equations to complete contains Abelian groups theory.

Example 3.13 . *During the completion of commutative rings theory modulo AG, the equation* $(x.y)+(x.0)=x.y$ *is generated. The orientation via symmetrisation produces the rule* $x.0\to 0$. *We see in this case that the symmetrisation technique includes in particular* cancellation. *Another equation generated during this completion is* $(x.y)+(x.(-y))=0$. *Symmetrisation gives directly the rule* $x.(-y)\to-(x.y)$, *without computing any AC critical pair, as in the usual AC completion.*

It is possible to apply the symmetrisation technique in normalised completion modulo commutative rings theory, boolean rings theory, and also to theories defining finite fields [27]. Unfortunately, there is no convergent system for fields theory, because of the conditional equation $x.x^{-1} = 1$ if $x \neq 0$.

4 Decidability of the word problem for some classes of equational theories

Now, we investigate termination issues of the completion process. It is already known that AC completion terminates when the initial set of equations is ground. Here we see how this result can be extended to S-normalised completion for some interesting S.

4.1 General results

We first look at some general results, true for arbitrary S. We assume that the ordering $\succ$ is total on ground terms and the initial equations are ground, this prevents completion from failure. We define the notion of *generator set* of a term. This extends Narendran and Rusinowitch's definition [29]. Let F be the set of symbols that appear in S.

Definition 4.1 *Let u be a (flat) term. The* generator set *of u (w.r.t F) is defined by $\gamma_F(u) = \{u\}$ if $\mathcal{H}\mathrm{ead}(u) \notin F$ and $\gamma_F(u) = \bigcup_{1 \leq i \leq n} \gamma_F(u_i)$ if $u = f(u_1, \ldots, u_n)$ with $f \in F$. Let E be a set of equations and R be a set of rules. The* generator set *of E and R, denoted $G_F(E, R)$, is the union of the generator sets of all members of equations of E and rules of R.*

We prove that along any derivation of the completion process, $G_F(E_n, R_n)$ cannot increase between two steps where E_n and R_n are completely simplified.

Proposition 4.2 *Let E_0 be a set of equations. Assume $E; R \overset{*}{\vdash} E'; R'$ is a sequence of derivations starting from $E_0; \emptyset$ and such that $E'; R'$ are no longer simplifyable, that is neither* Normalise, Delete, Simplify, Compose *nor* Collapse *can be applied. Then*

$$G_F(E, R) \succeq_{\mathrm{mul}} G_F(E', R')$$

Definition 4.3 *We say that the strategy* simplifies first *if the simplification rules* Normalise, Delete, Simplify, Compose *and* Collapse *have priority on* Orient *and* Deduce.

This condition on the strategy is essential. Otherwise, completion could diverge whereas R_∞ is finite indeed.

Proposition 4.4 *Assume that strategy simplifies first. Then if completion does not terminate, R_∞ is infinite and there are infinitely many rules such that the top symbol of their left-hand side is in F.*

In the following, termination is proven by showing that the inference rules Orient and Deduce do not increase $G_F(E, R)$. Then, assuming that completion does not terminate, we deduce that there are infinitely many rules for which the left-hand side is built from symbols of F and terms in $G_F(E, R)$.

4.2 Termination of completion

We first have to show how to define a total ordering in each case we are interested in. In the case of simple theories AC1, ACI, AC1I, AC0 and ACN, we can use the total AC ordering of Narendran and Rusinowitch [29] or the one of Nieuwenhuis and Rubio [30] (with the condition $+ > 0$ for ACN, in order to orient $x + x \rightarrow 0$).

In the case of Abelian groups, we can use a recursive path ordering with a total precedence of the form $\mathcal{F} > - > + > 0$, and such that $+$ has multi-set status and all other operators have lexicographic status (for totality).

In the case of commutative rings, it is a bit more complicated since the total AC ordering above always orient distributivity in the wrong way! The solution is to use the lexicographic extension of the *modified associative path ordering* [10] with precedence $1 > . > - > + > 0$, and then any total AC-compatible ordering [27]. Such an ordering can be used also for Boolean rings and finite fields theories.

Theorem 4.5 *If the initial set of equations is ground, then the S-normalised completion terminates when S is either AC, AC1, ACI, AC1I, AC0, ACN, AG, CR, BR or FF(p). As a consequence, every equational theory presented by $C \cup S$, where C is a set of ground equations and S is one of the previous theories, has a S-normalised canonical rewriting system, in particular it has a decidable word problem.*

	AC	AC1	AG	AG∪AC1	RRL	REVEAL
Computation time	26"2	22"9	2"9	3"0	4"9	22"6
Number of critical pairs generated	537	412	46	39	197	406

Figure 3: Commutative rings theory modulo AC, AC1 and AG

5 Some implementation results

5.1 Commutative rings modulo AC, AC1 and AG

We first show what happens when completing commutative rings theory modulo AC, AC1, AG and AG ∪ AC1. Figure 3 shows practical results and also compares with other AC completion systems RRL [18] and REVEAL [1]. We can see that completion modulo AC1, and moreover AG, are more efficient than AC completion. Our implementation is not as optimised as RRL and REVEAL hence AC completion is less efficient, but when completing modulo AG, it is more efficient indeed. The following example shows well why normalised completion is "optimised" w.r.t. AC completion: it is not only because some equations are already built in, it is also because new equations are inferred faster. When orienting the equation $x+(-x)=0$ in AC1-normalised completion, the equation $0+(-0)=0$ in Θ from which you obtain the rule $-0 \rightarrow 0$. In AC completion you need to compute some critical pairs to obtain this equation, that is to say you need AC unification to infer this new rule but not in AC1-completion. Note that this happens even if you still use AC unification, not AC1, as it is the case for the moment in our implementation. In the case of AG-normalised completion, the improvement is even more spectacular as mentionned in Example 3.13.

5.2 A canonical rewriting system for a finitely generated Abelian group

Consider the Abelian group G presented by $E = \{2a-3b+c=0, -3a+2b+3c=0, 2a+2b-2c=0\}$ [22]. We give the set of equations above to the AG-normalised completion algorithm, and the result is $\{b \rightarrow 9a, c \rightarrow 25a, 30a \rightarrow 0, -a \rightarrow 29a\}$. The AG-normalised completion of this system with our implementation takes 29" and computes only 14 critical pairs, whereas the AC completion of $E \cup$ AG by RRL takes 3'27" and computes 837 critical pairs, and by REVEAL takes 22" and computes 183 critical pairs. Of course, this shows the crucial role of symmetrisation.

5.3 Computation of a standard basis of a polynomial ideal

Now we show an example of standard basis computation using normalised completion. When polynomials have integer coefficients, computing a standard basis amounts to normalised completion modulo commutative rings theory.

Example 5.1 *To compute a standard basis of the ideal* $(2X^2Y-Y, 3XY^2-X)$ *over* $\mathbb{Z}$ *[16] we give to CR-normalised completion the set of equations* $\{2XXY-Y=0, 3XYY-X=0\}$ *where* X, Y *are two constants,* $Y > X$ *in the precedence. The completion will produce:*

$$\left\{\begin{array}{ll} 2XXY \rightarrow Y & 2XXYx \rightarrow Yx \\ -XXY \rightarrow XXY - Y & -XXYx \rightarrow XXYx - Yx \\ XXYY \rightarrow XX - YY & \\ 3YY \rightarrow 2XX & 3YYx \rightarrow 2XXx \\ -YY \rightarrow 2YY - 2XX & -YYx \rightarrow 2YYx - 2XXx \\ 2XXX \rightarrow X & 2XXXx \rightarrow Xx \\ -XXX \rightarrow XXX - X & -XXXx \rightarrow XXXx - Xx \end{array}\right.$$

which corresponds to the standard basis $\{2X^2Y-Y, X^2Y^2-X^2+Y^2, 3Y^2-2X^2, 2X^3-X\}$.

For polynomials with coefficients in a finite field, this can be done also by $FF(p)$-normalised completion where $FF(p)$ presents the finite field of cardinal p (it has already been remarked by Bündgen that computation of such a standard basis can be done by AC completion [9]).

The problem of embedding the computation of a standard basis of a polynomial ideal with coefficients in an infinite field like $\mathbb{Q}$, in an S-normalised completion for a well-chosen S remains open.

An interesting remark is that the termination result of the well-known algorithms for computing standard bases are particular cases of the termination result we have given.

6 Conclusion

Figure 4 shows, for various E, known results on decidability or undecidability of word problem of the

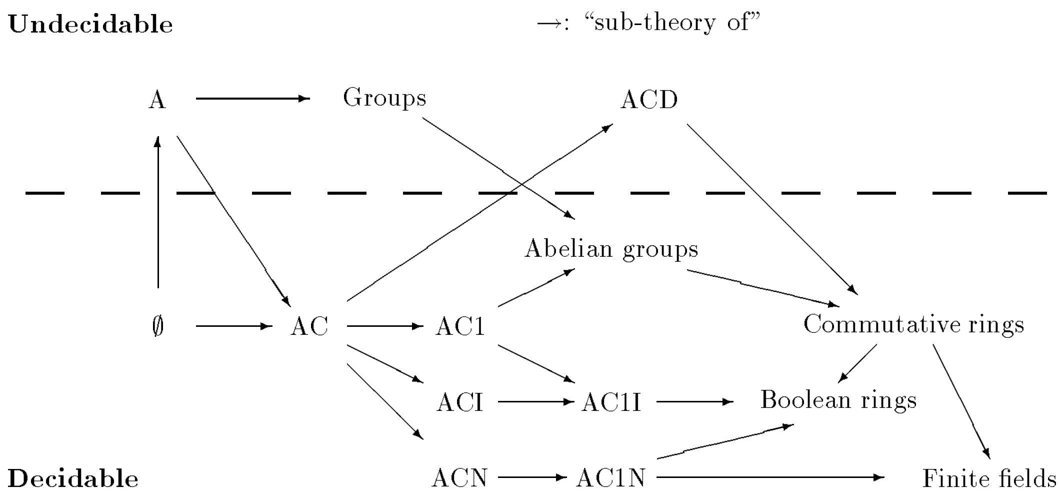

Figure 4: Decidability of the word problem of ground theories modulo E, for some E

classes of equational theories defined by E and an arbitrary set of ground equations.

In the cases where the word problem is decidable, this is a consequence of the termination of E-normalised completion, so the result is much stronger: every E-ground theory has a E-normalised rewrite system. The undecidability of word problem for ground theories modulo associativity was proved independently by Post and Markov in 1947 [28, 33], for groups theory it is a result of Novikov in 1955 [31, 34] and for ground theories modulo ACD it is a recent result [26].

As a conclusion, we have obtained theoretical results: the unification and the generalisation of decidability results, and a new completion algorithm, which generalises the already known completion modulo a theory. It also enjoys practical advantages: it needs an AC-compatible ordering only, not E-compatible, it allows to choose the most efficient unification algorithm, and allows in particular cases the use of precomputed and optimised normalising pairs (Θ, Ψ) of equations and rules. It has also the interesting property that it unifies Knuth-Bendix completion (and its extensions AC completion, AC1-constrained completion) and Buchberger's algorithm for computing standard bases.

Future work will be to find other interesting particular theories, like non-commutative groups. To solve the problem of fields theory, it may be interesting to see if we can use a conditional rewrite system for S. From a practical point of view, it remains to check whether using AC1 or ACI unification is really interesting (our implementation uses only AC unification). We also have to study whether the well-known critical pair criteria can be applied to normalised completion.

Acknowledgements

I'd like to thank Jean-Pierre Jouannaud and Hubert Comon for their useful comments about the preliminary version of this paper.

References

[1] S. Anantharaman. REVEAL: a users' guide. Rapport de Recherche, Laboratoire d'Informatique Fondamentale d'Orléans, 1993.

[2] L. Bachmair. *Canonical Equational Proofs.* Birkhäuser, Boston, 1991.

[3] L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. *Theoretical Comput. Sci.*, 67(2&3):173–201, Oct. 1989.

[4] L. Bachmair, N. Dershowitz, and J. Hsiang. Orderings for equational proofs. In *Proc. 1st IEEE Symp. Logic in Computer Science, Cambridge, Mass.*, pages 346–357, June 1986.

[5] L. Bachmair and D. A. Plaisted. Termination orderings for associative-commutative rewriting systems. *Journal of Symbolic Computation*, 1(4):329–349, Dec. 1985.

[6] T. Baird, G. Peterson, and R. Wilkerson. Complete sets of reductions modulo Associativity, Commutativity and Identity. In *Proc. 3rd Rewriting Techniques and Applications, Chapel Hill, LNCS 355*, pages 29–44. Springer-Verlag, Apr. 1989.

[7] A. Ben Cherifa and P. Lescanne. An actual implementation of a procedure that mechanically proves

termination of rewriting systems based on inequalities between polynomial interpretations. In *Proc. 8th Int. Conf. on Automated Deduction, Oxford, England, LNCS 230*, pages 42–51. Springer-Verlag, July 1986.

[8] G. Birkhoff. On the structure of abstract algebras. In *Proc. Cambridge Phil. Society, 31*, 1935.

[9] R. Bündgen. Simulating Buchberger's algorithm by a Knuth-Bendix completion procedure. In R. V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, LNCS 488*, Como, Italy, Apr. 1991. Springer-Verlag.

[10] C. Delor and L. Puel. Extension of the associative path ordering to a chain of associative-commutative symbols. In *Proc. 5th Rewriting Techniques and Applications, Montréal, LNCS 690*, 1993.

[11] N. Dershowitz. Termination of rewriting. *Journal of Symbolic Computation*, 3(1):69–115, Feb. 1987.

[12] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 243–309. North-Holland, 1990.

[13] E. Domenjoud. Outils pour la déduction automatique dans les théories associatives-commutatives. Thèse de doctorat de l'université de Nancy I, 1991.

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

[15] J.-P. Jouannaud and C. Marché. Termination and completion modulo associativity, commutativity and identity. *Theoretical Comput. Sci.*, 104:29–51, 1992.

[16] A. Kandri-Rody and D. Kapur. An algorithm for computing the Gröbner basis of a polynomial ideal over an Euclidean ring. Technical Report 84CRD045, CRD, General Electric Company, Schenectady, New-York, Dec. 1984.

[17] D. Kapur, D. Musser, and P. Narendran. Only prime superpositions need be considered for the Knuth-Bendix procedure. *Journal of Symbolic Computation*, 4:19–36, 1988.

[18] D. Kapur and H. Zhang. An overview of the rewrite rule laboratory (RRL). In *Proc. 3rd Rewriting Techniques and Applications, Chapel Hill, LNCS 355*, pages 559–563. Springer-Verlag, 1989.

[19] C. Kirchner, editor. *Unification*. Academic Press, 1990.

[20] C. Kirchner and H. Kirchner. Constraint equational reasoning. In *Proc. of the 20th Int. Symp. on Symbolic and Algebraic Computation, Portland, Oregon*, 1989.

[21] 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.

[22] D. Lankford, G. Butler, and A. Ballantyne. A progress report on new decision algorithms for finitely presented abelian groups. In *Proc. 7th Int. Conf. on Automated Deduction, Napa, LNCS 170*. Springer-Verlag, May 1984.

[23] D. S. Lankford and A. M. Ballantyne. Decision procedures for simple equational theories with commutative-associative axioms: Complete sets of commutative-associative reductions. Research Report Memo ATP-39, Department of Mathematics and Computer Science, University of Texas, Austin, Texas, USA, Aug. 1977.

[24] P. Le Chenadec. *Canonical forms in finitely presented algebras*. Pitman, London, 1986.

[25] C. Marché. On ground AC-completion. In R. V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, LNCS 488*, Como, Italy, Apr. 1991. Springer-Verlag.

[26] C. Marché. The word problem of ACD-ground theories is undecidable. *International Journal of Foundations of Computer Science*, 3(1):81–92, 1992.

[27] C. Marché. Réécriture modulo une théorie présentée par un système convergent et décidabilité des problèmes du mot dans certains classes de théories équationnelles. Thèse de Doctorat, Université de Paris-Sud, France, 1993.

[28] A. A. Markov. On the impossibility of certain algorithms in the theory of associative systems. *Dokl. Akad. Nauk SSSR*, 55(7):587–590, 1947. In Russian, English translation in C.R. Acad. Sci. URSS, 55, 533-586.

[29] P. Narendran and M. Rusinowitch. Any ground associative-commutative theory has a finite canonical system. In R. V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, LNCS 488*, Como, Italy, Apr. 1991. Springer-Verlag.

[30] R. Nieuwenhuis and A. Rubio. A precedence-based total AC-compatible ordering. In C. Kirchner, editor, *Proc. 5th Rewriting Techniques and Applications, Montréal, LNCS 690*. Springer-Verlag, June 1993.

[31] P. S. Novikov. On the algorithmic unsolvability of the word problem in group theory. *Trudy Mat. Inst. Steklov*, 44:1–143, 1955. in Russian.

[32] G. E. Peterson and M. E. Stickel. Complete sets of reductions for some equational theories. *J. ACM*, 28(2):233–264, Apr. 1981.

[33] E. L. Post. Recursive unsolvability of a problem of Thue. *Journal of Symbolic Logic*, 13:1–11, 1947.

[34] J. Stillwell. The word problem and the isomorphism problem for groups. *Bulletin of the American Mathematical Society*, 6(1):33–56, Jan. 1982.