J. Symbolic Computation (1996) **11**, 1–000

Normalized Rewriting: an alternative to Rewriting modulo a Set of Equations†

Claude Marché

LRI – URA 410 du CNRS, Université de Paris-sud, F-91405 Orsay cedex, France

(Received 1 February 1996)

In a first part, we introduce *normalized rewriting*, a new rewrite relation. It generalizes former notions of rewriting modulo a set of equations E, dropping some conditions on E. For example, E can now be the theory of identity, idempotence, the theory of Abelian groups or the theory of commutative rings. We give a new completion algorithm for normalized rewriting. It contains as an instance the usual AC completion algorithm, but also the well-known Buchberger algorithm for computing Gröbner bases of polynomial ideals.

In a second part, 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 theories E. As a consequence, we obtain the decidability of the word problem for some classes of equational theories, including the AC-ground case (a result known since 1991), the ACUI-ground case (a new result to our knowledge), and the cases of ground equations modulo the theory of Abelian groups and commutative rings, which is already known when the signature contains only constants, but is new otherwise.

Finally, we give implementation results which show the efficiency of normalized 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 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 orienting the equations into rewrite rules via the ordering. To transform an equational proof into a rewrite proof one must replace 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 (Knuth and Bendix, 1970). In completion, the axioms used are therefore in a constant state of flux; these changes

† This work is partly supported by the "GDR Programmation du CNRS", the ESPRIT Working Group "Compass" and the ESPRIT Basic Research Action "TYPES"

0747–7171/90/000000 + 00 $03.00/0

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. This difficulty has been resolved (Lankford and Ballantyne, 1977; Peterson and Stickel, 1981) by building associativity and commutativity in the rewriting process via AC pattern matching: rewriting modulo AC; as well as in the computation of AC critical pairs via AC-unification: completion modulo AC. This has been further generalized to rewriting modulo E and completion modulo E for an arbitrary equational theory E provided E-unification is finitary and the subterm ordering modulo E is well-founded (Bachmair and Dershowitz, 1989; Jouannaud and Kirchner, 1986).

This technique excludes therefore some important sets of axioms like the identity law ($x + 0 = x$ where $+$ is AC, denoted ACU), group theory, commutative ring theory, idempotence ($x + x = x$, denoted ACI), etc, from being part of the set E. Indeed, in all these cases, E-rewriting does not terminate in general. For example, rewriting modulo ACU yields the following infinite derivation, using the rule $-(x + y) \to (-x) + (-y)$ for computing the inverse of a sum:

$$\begin{array}{lll} -0 & =_{\mathrm{ACU}} -(0+0) & \to (-0)+(-0) \\ & =_{\mathrm{ACU}} -(0+0)+(-0) & \text{etc...} \end{array}$$

It is possible to overcome this difficulty using *constrained* rewriting (Kirchner and Kirchner, 1989): Baird, Peterson and Wilkerson (Baird et al., 1989) have described a constrained rewriting technique specialized to ACU, and Jouannaud and Marché (Jouannaud and Marché, 1992) have described a corresponding completion algorithm modulo ACU. Unfortunately, their approach does nor work for other theories mentioned above: for any rule $l \to r$, taking ACI for E yields:

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

and taking now AG (Abelian group theory) for E yields:

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

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

Independently of all this, in the field of computational algebra, an algorithm has been proposed by Buchberger in 1965 (Buchberger, 1965) for computing *Gröbner bases* of polynomial ideals. Although Knuth-Bendix completion has been described independently, it appears that both algorithms proceed similarly, by this technique of orienting equations into rules and deduction between two rules by critical pair computation. This similarity has first been mentioned by Loos and Buchberger (Loos, 1981; Buchberger and Loos, 1982). Since that time, many researchers have tried to unify in some sense these algorithms. We can cite the works (Le Chenadec, 1986; Kandri-Rody et al., 1989; Pottier, 1989; Bündgen, 1991a; Bündgen, 1991b; Bachmair and Ganzinger, 1994) and we miss certainly many others.

The main motivation of this paper is to make the similarity between Knuth-Bendix completion and the Buchberger algorithm explicit, by describing a general algorithm called *S-normalized completion* where S is a parameter, such that both algorithms are instances of this general algorithm for a particular choice of S. This has been achieved in two steps.

The first step is the modification of the rewrite relation. We want to view reduction of polynomials as rewriting on the algebra of polynomials, that is rewriting modulo commutative ring theory. As remarked before, this is not possible by the technique of Jouannaud and Kirchner (Bachmair and Dershowitz, 1989; Jouannaud and Kirchner, 1986). To solve this problem, we define a new rewrite relation, inspiring ourselves by the way polynomials are reduced in Buchberger algorithm: the distributive law and other polynomial laws are applied *prior* to other polynomial reductions. This is done in Section 2 where we define precisely what normalized rewriting is, and we show that the termination of this new rewrite relation can be checked by a reduction ordering that need not be compatible with E (such orderings do not exist in general) like the former rewriting modulo E (but still has to be compatible with AC if there are AC operators).

The second step is the generalization of the process of orientation. In Buchberger algorithm, when turning a polynomial into a reduction rule, one selects the *head monomial* of the polynomial which becomes the left-hand side of the rule. This has been taken into account in S-normalized completion by using the *symmetrization* technique proposed by Le Chenadec (Le Chenadec, 1986) generalized to non-ground (i.e. with variables) equations. Furthermore, this generalization requires adding new equations called *critical instances* as remarked before for ACU-constrained completion (Jouannaud and Marché, 1992), and leads to the notion of *normalizing pairs*. This is done in Section 3.

One interesting remark about the comparison of Buchberger algorithm and Knuth-Bendix completion is that Buchberger algorithm always terminates, whereas Knuth-Bendix completion may loop infinitely. The reason is that when viewing Buchberger algorithm as a completion procedure, equations to be completed are *ground*, and it is known that Knuth-Bendix completion and furthermore AC-completion (Narendran and Rusinowitch, 1991; Marché, 1991) terminate if the initial set of equations is ground. Now, the interesting question is whether the results of termination of Buchberger algorithm and ground AC-completion are in fact the same result of termination of ground S-normalized completion. Unfortunately, the answer is no: ground S-normalized completion does not terminate in general, but only for particular S. The interesting point is that we can show some general results (true for any S) about ground S-normalized completion, which can be used to prove by simple arguments the termination in the cases of S we are interested in. This method makes the harder part of the termination proofs of Buchberger algorithm and ground AC-completion common. This is done in Section 5. We obtain then 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.

Finally, we give in section 6 some examples of normalized completion obtained by our implementation C*i*ME†, and we show in particular some interesting benchmarks and also how normalized completion can be used to compute Gröbner bases of polynomial ideals.

† `http://www.lri.fr/~marche/cime.html` or `ftp://lri.lri.fr/LRI/soft/cime/cime.tar.Z`

2. Normalized rewriting

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

2.1. Basic definitions

Our notations and definitions are consistent with those given in the survey of Dershowitz and Jouannaud (Dershowitz and Jouannaud, 1990).

We denote $\mathcal{T}(\mathcal{F},\mathcal{X})$, or $\mathcal{T}$ for short, as 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 as the set of positions and non variable positions of a term s. We denote Λ the top position. Two incomparable positions (*i.e.* none is a prefix of the other) are said to be *parallel* and denoted $p \parallel q$. The subterm 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 theorem (Birkhoff, 1935): $s = t$ 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.).

Congruence classes modulo AC can be represented as *flat* terms, that is terms of the form $(x+y)+z$ and $x+(y+z)$ are flattened to $x+y+z$. 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. 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 subterms of AC symbols.

We say that two terms s and t are unifyable 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 (Kirchner, 1990). We denote by $\mathrm{CSU}_E(s,t)$ a complete set of E-unifiers of s and t.

We denote $\trianglerighteq$ as the subterm ordering, and $\mathrel{\cdot\!\!\trianglerighteq}$ as the *encompassment* ordering, that is $s \mathrel{\cdot\!\!\trianglerighteq} t$ if there exist p and σ such that $s|_p = t\sigma$. These definitions are assumed to be modulo AC (*i.e.* defined on flat terms) if there are AC operators. We remark that both orderings are well-founded, and that $s \mathrel{\cdot\!\!\triangleright} t$ if and only if $p \neq \Lambda$ or $p = \Lambda$ and σ is not a renaming.

2.2. AC-rewriting on flat terms

Formal and complete definitions of flattening and rewriting on flat terms can be found in (Domenjoud, 1991; Kapur et al., 1988; Marché, 1993).

We use rewriting on flat terms, that is we say that s *rewrites* to t by $l \to r$ at position

$p \in \mathcal{FP}\mathrm{os}(s)$, denoted as

$$s \xrightarrow[l \to r]{p} t,$$

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" (Peterson and Stickel, 1981): 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 generalize 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 $a+b \to d$ and $a + c \to e$ 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 generalized notion of overlapping.

From now on, we assume we have a signature containing arbitrarily many AC operators (possibly none), and everything is implicitly modulo the associative-commutative theory defined by these operators. When R is a set of rules, we denote by $\to_R$ the rewrite relation by R (implicitly modulo AC as said before). R is said to be convergent if $\to_R$ is well-founded and confluent.

2.3. Definition of normalized rewriting

We assume we are given an algebra by a signature and a set of equations defining some relations between constructors or operators. Our goal is to define a rewriting relation on this quotient algebra. Let us assume this equational theory has an equivalent convergent rewriting system S.

Definition 2.1. *Let us denote by $s{\downarrow_S}$ the S-normal form of a term s. The S-*normalized rewrite relation*, denoted as*

$$s \xrightarrow[l \to r/S]{p} t,$$

is defined by

$$s' = s{\downarrow_S} \text{ and } s' \xrightarrow[l \to r]{p} t.$$

When R is a set of rules, we denote by $\to_{R/S}$ the S-normalized rewrite relation by R.

Example 2.2. *Assume we have an algebra containing an AC operator $+$ which has a unit 0. We take then $S = \mathrm{ACU}(+, 0) = \{x + 0 \to x\}$. Assume now we would like to rewrite by $R = \{-(x + y) \to (-x) + (-y)\}$. We have $-(a + b) \to_{R/S} (-a) + (-b)$ 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 normalized rewriting captures the notion of ACU-constrained rewriting (Baird et al., 1989; Jouannaud and Marché, 1992).

EXAMPLE 2.3. *Assume now our algebra is a ring of multivariate polynomials. We take for S the convergent rewrite system of commutative rings theory, that is*

$$\left\{\begin{array}{ll} x+0 \rightarrow x & x+(-x) \rightarrow 0 \\ -0 \rightarrow 0 & -(-x) \rightarrow x \\ -(x+y) \rightarrow (-x)+(-y) & x * 1 \rightarrow x \\ x *(y+z) \rightarrow (x * y)+(x * z) & x * 0 \rightarrow 0 \\ x *(-y) \rightarrow -(x * y) & \end{array}\right.$$

Assume now we have $R=\{X * X \rightarrow Y\}$ where X and Y are some constants (representing indeterminates, hence terms represent polynomials). Then $X * X * X \rightarrow_{R/S} X * Y$ 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 normalized rewriting captures the notion of polynomial reduction used in Gröbner basis computation, where the distributivity law is applied before the rules.

2.4. Termination of normalized rewriting

Proving termination of (usual) rewriting modulo an equational theory E requires an ordering compatible with E. Such an ordering does not exist in general, for example there is no reduction ordering compatible with idempotence 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 (Dershowitz, 1987). For definitions of AC-compatible orderings, see (Bachmair and Plaisted, 1985; Ben Cherifa and Lescanne, 1986; Narendran and Rusinowitch, 1991; Nieuwenhuis and Rubio, 1993; Delor and Puel, 1993).

From now on, we assume we are given an AC reduction ordering $\succeq$, such that the set of rules S satisfies $\rightarrow_S \subseteq \succ$ (that is the termination of S 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-normalized rewrite relation $\rightarrow_{R/S}$ is well-founded.*

3. Normalized completion

We still assume we are given a signature containing arbitrarily many AC operators, a set of equations which has an equivalent convergent rewriting system S, together with a reduction ordering $\succeq$ such that $\rightarrow_S \subseteq \succ$. We assume now we have a new set of equations E_0 defining for example some new operations. Our aim is to complete E_0 into a set of rules R such that S-normalized rewriting by R is well-founded and confluent.

In Section 3.2, we give a set of inference rules for completing a set of equations into a normalized rewrite system. The completeness is proved by the now customary normalization proof method (Bachmair and Dershowitz, 1989; Bachmair et al., 1986). We first need to introduce in Section 3.1 the notion of *normalizing pairs*, which in some sense will replace the usual notion of orientation in completion procedures.

3.1. Proof algebra and normalizing pairs

This section is organized in four parts: first we define formally the proof algebra, second we give an ordering on this algebra, third we investigate some commutation properties of the algebra, and fourth we define the notion of normalizing pairs.

3.1.1. Proof algebra

We assume here that all terms are flattened with respect to AC symbols.

Definition 3.1. *The algebra of equational proofs is generated by the elementary proofs:*

(i) AC step: $s \underset{\mathrm{AC}}{\longleftrightarrow} t$ *if* $s \equiv t$;

(ii) equational step: $s \overset{\sigma,p}{\underset{l=r}{\longleftrightarrow}} t$ *where* $s|_p = l\sigma$ *and* $t = s[r\sigma]_p$ *for an equation* $l = r$;

(iii) rewrite step: $s \overset{\sigma,p}{\underset{l\to r}{\longrightarrow}} t$ *where* $s|_p = l\sigma$ *and* $t = s[r\sigma]_p$ *for a rule* $l \to r$;

(iv) S-normalizing step: $s \xrightarrow{S} t$ *if* $s \to_S t$;

and the concatenation of proofs, denoted P.Q, where the last term of P is assumed to equal the first of Q. Concatenation is implicitly associative. We say that a proof is in $E \cup R$ *(or in* an $E;R$-proof*) if its equations (resp. its rules) are in E (resp. in R).*

We will also write

$$s \xrightarrow[S]{(\geq p)^*} t$$

to denote a sequence of S-reductions at position p or below.

In the following, we need to consider the symmetric proof of proof P defined by:

$$\begin{aligned}
sym(s \underset{\mathrm{AC}}{\longleftrightarrow} t) &= t \underset{\mathrm{AC}}{\longleftrightarrow} s \\
sym(s \overset{\sigma,p}{\underset{l=r}{\longleftrightarrow}} t) &= t \overset{\sigma,p}{\underset{r=l}{\longleftrightarrow}} s \\
sym(s \overset{\sigma,p}{\underset{l\to r}{\longrightarrow}} t) &= t \overset{\sigma,p}{\underset{l\to r}{\longleftarrow}} s \\
sym(s \xrightarrow{S} t) &= t \xleftarrow{S} s \\
sym(P.Q) &= sym(Q).sym(P)
\end{aligned}$$

3.1.2. Proof reduction ordering

For proving completeness of the completion algorithm, we need the proof algebra to be ordered in a convenient way. Unusually, we define the ordering we will use before describing the completion itself, because the definition of normalizing depends on this ordering.

Let $s{\downarrow_p}$ be the result of S-normalizing 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 $\{s,t\}$ otherwise.

NORMALIZATION OF SUBSTITUTIONS

$$s \underset{u=v}{\overset{\sigma,p}{\longleftrightarrow}} t \Longrightarrow s \overset{(\geq p)*}{\xrightarrow{S}} s' \underset{u=v}{\overset{\sigma',p}{\longleftrightarrow}} t' \overset{(\geq p)*}{\xleftarrow{S}} t \quad \text{if } \sigma' = \sigma{\downarrow_S}.$$

NORMALIZATION OF EQUATIONS

$$s \underset{u=v}{\overset{\sigma,p}{\longleftrightarrow}} t \Longrightarrow s \overset{(\geq p)*}{\xrightarrow{S}} s' \underset{u'=v'}{\overset{\sigma,p}{\longleftrightarrow}} t' \overset{(\geq p)*}{\xleftarrow{S}} t \quad \text{if } u' = u{\downarrow_S} \text{ and } v' = v{\downarrow_S}.$$

EQUATIONAL STEP NORMALIZATION AT A PARALLEL POSITION

$$s \underset{u=v}{\overset{\sigma,p}{\longleftrightarrow}} t \Longrightarrow s \underset{l\to r}{\overset{q,\sigma}{\xrightarrow{S}}} s' \underset{u=v}{\overset{p,\sigma}{\longleftrightarrow}} t' \underset{l\to r}{\overset{q,\sigma}{\xleftarrow{S}}} t \quad \text{if } l \to r \in S \text{ and } p \parallel q.$$

EQUATIONAL STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION

$$s \underset{u=v}{\overset{\sigma,p}{\longleftrightarrow}} t \Longrightarrow s \underset{l\to r}{\overset{\sigma,q}{\xrightarrow{S}}} s' \left(\underset{u=v}{\overset{\sigma,q.o'}{\longleftrightarrow}} \right)^*_{o'\in P'} s'' \underset{l\to r}{\overset{\sigma',q}{\xleftarrow{S}}} t' \left(\underset{v=u}{\overset{\sigma,q.o}{\longleftrightarrow}} \right)^*_{o\in P} t$$
$$\text{if } s = s{\downarrow_p},\ u \succ v,\ l \to r \in S,\ p = qp' \text{ with } p' \notin \mathcal{FP}\text{os}(l),\ p' = q'p'' \text{ with } l|_{q'} = x \in \mathcal{X},$$
$$P = \{o \in \mathcal{P}\text{os}(l) | l|_o = x\} - \{q'\},\ P' = \{o' \in \mathcal{P}\text{os}(r) | r|_{o'} = x\},\ \sigma' : \begin{cases} x\sigma' = x\sigma[v\sigma]_{p''} \\ \text{if } y \neq x,\ y\sigma' = y\sigma \end{cases}$$

REWRITE STEP NORMALIZATION AT A PARALLEL POSITION

$$s \underset{u\to v}{\overset{\sigma,p}{\longrightarrow}} t \Longrightarrow s \underset{l\to r}{\overset{q,\sigma}{\xrightarrow{S}}} s' \underset{u\to v}{\overset{p,\sigma}{\longrightarrow}} t' \underset{l\to r}{\overset{q,\sigma}{\xleftarrow{S}}} t \quad \text{if } l \to r \in S \text{ and } p \parallel q.$$

REWRITE STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION

$$s \underset{u\to v}{\overset{\sigma,p}{\longrightarrow}} t \Longrightarrow s \underset{l\to r}{\overset{\sigma,q}{\xrightarrow{S}}} s' \left(\underset{u\to v}{\overset{\sigma,q.o'}{\longrightarrow}} \right)^*_{o'\in P'} s'' \underset{l\to r}{\overset{\sigma',q}{\xleftarrow{S}}} t' \left(\underset{u\to v}{\overset{\sigma,q.o}{\longleftarrow}} \right)^*_{o\in P} t$$
$$\text{if } s = s{\downarrow_p},\ l \to r \in S,\ p = qp' \text{ with } p' \notin \mathcal{FP}\text{os}(l),\ p' = q'p'' \text{ with } l|_{q'} = x \in \mathcal{X},$$
$$P = \{o \in \mathcal{P}\text{os}(l) | l|_o = x\} - \{q'\},\ P' = \{o' \in \mathcal{P}\text{os}(r) | r|_{o'} = x\},\ \sigma' : \begin{cases} x\sigma' = x\sigma[v\sigma]_{p''} \\ \text{if } y \neq x,\ y\sigma' = y\sigma \end{cases}$$

Figure 1. Proof reduction: S-normalizations

DEFINITION 3.2. *The ordering $\succ_{\mathcal{P}}$ on proofs is defined as follows: the* complexity *of a proof is the multiset of the complexities of its elementary subproofs, defined by*

$$\begin{array}{rcl} C(s \underset{\mathrm{AC}}{\longleftrightarrow} t) & = & \langle \{\perp, \{s\}, \perp, \perp\rangle \\ C(s \underset{l=r}{\overset{\sigma,p}{\longleftrightarrow}} t) & = & \langle \{s{\downarrow_p}, t{\downarrow_p}\}, \{s,t\}, \perp, \perp\rangle \\ C(s \underset{l\to r}{\overset{\sigma,p}{\longrightarrow}} t) & = & \langle c(s,p,t), \{s\}, l, r\sigma\rangle \\ C(s \xrightarrow{S} t) & = & \langle \perp, \{s\}, \perp, \perp\rangle \end{array}$$

where $\perp$ is a new element. Two elementary complexities are compared in the lexicographic extension of the orderings $\succeq_{mul}$ for the first and second components, encompassment ordering $\trianglerighteq$ for the third, $\succeq$ for the fourth. $\perp$ is always considered as minimal. Proofs are compared by their complexities in the multiset extension of the ordering above.

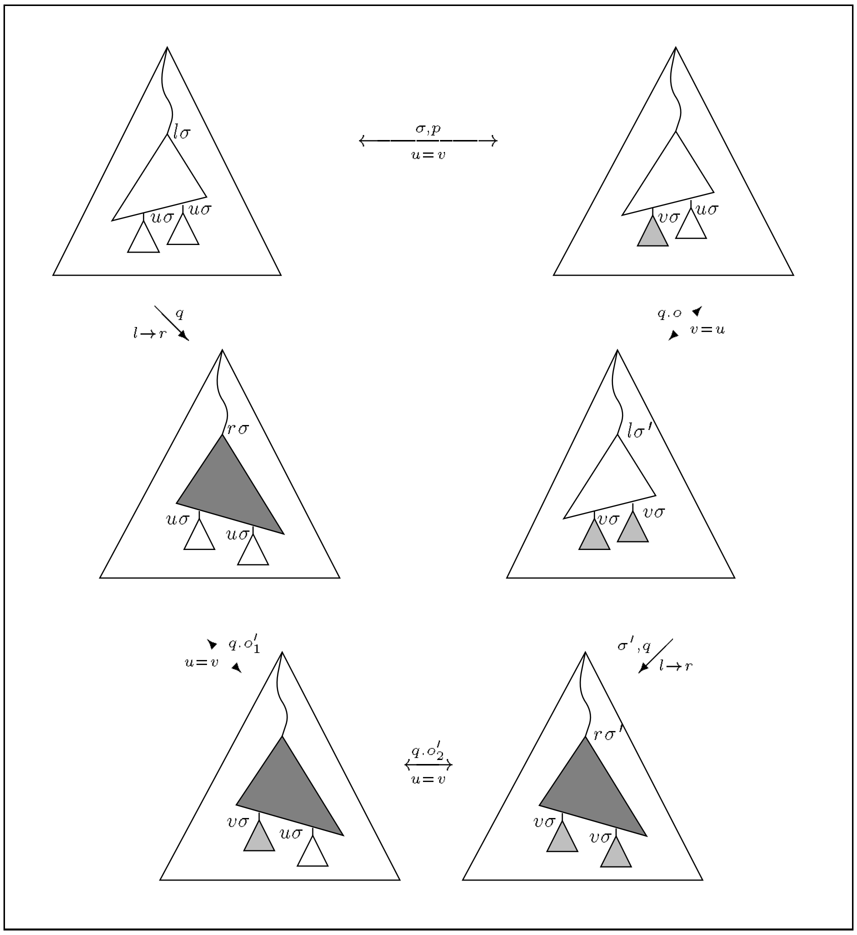

Figure 2. Equational step normalization at a variable-superposing position

Notice that $C(sym(P)) \equiv C(P)$, that is two symmetric proofs have the same complexity (modulo AC).

LEMMA 3.3. *The ordering $\succ_{\mathcal{P}}$ on the algebra of equational proofs is well-founded and monotonic: if $P \succ_{\mathcal{P}} P'$ then $Q.P.R \succ_{\mathcal{P}} Q.P'.R$.*

PROOF. $\succ_{\mathcal{P}}$ is well-founded since it is built up from well-founded orderings with the functionals *lex* and *mul* which preserve well-foundedness, and monotonicity follows from monotonicity of the multiset functional. □

We remark that this ordering is *not* stable by instantiation, since being in S-normal form is not, but this is not a required property for our purpose (see the definition of proof reduction below).

S-CONFLUENCE

$$s \xleftarrow[S]{+} t \xrightarrow[S]{+} u \Longrightarrow s \xrightarrow[S]{*} s' \xleftrightarrow[\mathrm{AC}]{*} u' \xleftarrow[S]{*} u$$

COMMUTATION OF A DISJOINT PEAK

$$s \xleftarrow[l_1 \to r_1]{\sigma,p} t \xrightarrow[l_2 \to r_2]{\sigma,q} u \Longrightarrow s \xrightarrow[l_2 \to r_2]{\sigma,q} v \xleftarrow[l_1 \to r_1]{\sigma,p} u \quad \text{if } p \parallel q.$$

COMMUTATION OF A VARIABLE-SUPERPOSING PEAK

$$s \xleftarrow[l_1 \to r_1]{\sigma,p} t \xrightarrow[l_2 \to r_2]{\sigma,pq} u \Longrightarrow s \left(\xrightarrow[l_2 \to r_2]{\sigma,p.o} \right)_{o \in P} s' \xleftarrow[l_1 \to r_1]{\sigma',p} v \xleftrightarrow[\equiv]{*} u' \left(\xleftarrow[l_2 \to r_2]{\sigma,p.o'} \right)_{o' \in P'} u$$
$$\text{if } q \notin \mathcal{FP}\mathrm{os}(r_1),\ q = p'q' \text{ with } r_1|_{p'} = x \in \mathcal{X},\ P = \{o \in \mathcal{P}\mathrm{os}(r_1) | r_1|_o = x\},$$
$$P' = \{o' \in \mathcal{P}\mathrm{os}(l_1) | l_1|_{o'} = x\},\ \sigma' : \begin{cases} x\sigma' = x\sigma[r_2\sigma]_{q'} \\ \text{if } y \neq x,\ y\sigma' = y\sigma \end{cases}$$

Figure 3. Proof reduction: commutation of non-critical peaks

3.1.3. Commutation properties of the proof algebra

The algebra of proofs satisfies some commutation properties which allows one to reduce proofs. This reduction of proofs will be defined from a set of proof reduction rules by taking the symmetric (if $P \Longrightarrow Q$ then $sym(P) \Longrightarrow sym(Q)$) and monotonic closure (but not closure by instantiation, that's why we don't need $\succ_{\mathcal{P}}$ being stable by instantiation).

Figure 1 shows how to reduce equational steps and rewrite steps by S-normalization. Notice that the rule EQUATIONAL STEP NORMALIZATION AT VARIABLE-SUPERPOSING POSITION applies only if $u \succ v$; if $u \prec v$ the symmetrical rule applies, and nothing applies if they are incomparable. Figure 2 displays graphically what happens in rule EQUATIONAL STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION and analogously in rule REWRITE STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION, white parts representing l and u, grey parts representing v and black parts representing r. Figure 3 shows how to reduce proofs in "peak" pattern.

The following lemma shows why we can talk about these rules as proof reduction rules.

LEMMA 3.4. *Proof reduction by commutation is well-founded.*

PROOF. All the commutation rules above decrease with respect to $\succ_{\mathcal{P}}$: verifications are made on Figure 4; and $\succ_{\mathcal{P}}$ is stable by symmetry and is monotonic. □

In particular, any proof possesses a normal form (not necessarily unique), and we call *irreducible* proofs the proofs in normal form.

3.1.4. Normalizing pairs

There are still some undesirable proofs patterns that are irreducible by the commutation rules above:

$$s \xleftrightarrow[u=v]{\sigma,p} t \text{ and } s \xrightarrow[u \to v]{\sigma,p} t$$

rule $P \Longrightarrow Q$	$C(P)$	$C(Q)$	proof of $P \succ_P Q$
NORMALIZATION OF SUBSTITUTIONS	$\langle \{s{\downarrow_p}, t{\downarrow_p}\},$ $\{s,t\},\ldots\rangle$	$\langle \{s'{\downarrow_p}, t'{\downarrow_p}\},$ $\{s',t'\},\ldots\rangle$ $\langle \bot,\ldots\rangle^*$	same first component, $s \succ s'$ and $t \succ t'$
NORMALIZATION OF EQUATIONS	$\langle \{s{\downarrow_p}, t{\downarrow_p}\},$ $\{s,t\},\ldots\rangle$	$\langle \{s'{\downarrow_p}, t'{\downarrow_p}\},$ $\{s',t'\},\ldots\rangle$ $\langle \bot,\ldots\rangle^*$	same first component, $s \succ s'$ and $t \succ t'$
EQUATIONAL STEP NORMALIZATION AT A PARALLEL POSITION	$\langle \{s{\downarrow_p}, t{\downarrow_p}\},\ldots\rangle$	$\langle \{s'{\downarrow_p}, t'{\downarrow_p}\},\ldots\rangle$ $\langle \bot,\ldots\rangle^*$	$s{\downarrow_p} \succ s'{\downarrow_p}$ and $t{\downarrow_p} \succ t'{\downarrow_p}$ since S-reduction is done at a position parallel to p
EQUATIONAL STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION	$\langle \{s{\downarrow_p}, t{\downarrow_p}\},\ldots\rangle$	$\langle \{s'{\downarrow_o}, s''{\downarrow_o}\},\ldots\rangle^*$ $\langle \{t'{\downarrow_{o'}}, t{\downarrow_{o'}}\},\ldots\rangle^*$ $\langle \bot,\ldots\rangle^*$	since $s\|_p = u\sigma$ is S-irreducible, $s'{\downarrow_o} = s'$, $s''{\downarrow_o} = s''$, $t'{\downarrow_{o'}} = t'$ and $t{\downarrow_{o'}} = t$, and $u \succ v$.
REWRITE STEP NORMALIZATION AT A PARALLEL POSITION	$\langle c(s,p,t),\ldots\rangle$	$\langle c(s',p,t),\ldots\rangle$ $\langle \bot,\ldots\rangle^*$	$c(s,p,t) \succ_{mul} c(s',p,t)$ since S-reduction is done at a position parallel to p, hence $s = s{\downarrow_p}$ iff $s' = s'{\downarrow_p}$
REWRITE STEP NORMALIZATION AT A VARIABLE-SUPERPOSING POSITION	$\langle c(s,p,t),\ldots\rangle$	$\langle c(s',o,s''),\ldots\rangle^*$ $\langle c(t,o',t'),\ldots\rangle^*$ $\langle \bot,\ldots\rangle^*$	since $s\|_p = u\sigma$ is S-irreducible, $c(s,p,t) = \{s\}$, $c(s',o,s'') = \{s'\}$ and $c(t,o',t') = \{t\}$; and $u \succ v$.
S-CONFLUENCE	$\langle \bot, \{t\},\ldots\rangle$ $\langle \bot, \{t\},\ldots\rangle$	$\langle \bot, \{s\},\ldots\rangle$ $\langle \bot, \{u\},\ldots\rangle$	$t \succ s$ and $t \succ u$
COMMUTATION OF A DISJOINT PEAK	$\langle c(t,p,s),\ldots\rangle$ $\langle c(t,q,u),\ldots\rangle$	$\langle c(s,q,v),\ldots\rangle$ $\langle c(u,p,v),\ldots\rangle$	$t \succ s$, $t \succ v$ hence $c(t,p,s) \succeq_{\mathrm{mul}} \{t\} \succ_{\mathrm{mul}} \{s,v\} \succeq_{\mathrm{mul}} c(s,q,v)$ and analogously $c(t,q,u) \succ_{\mathrm{mul}} c(u,q,v)$.
COMMUTATION OF A VARIABLE-SUPERPOSING PEAK	$\langle c(t,p,s),\ldots\rangle,$ $\langle c(t,pq,u),\ldots\rangle$	$\langle \{\ldots\},\ldots\rangle^*,$ $\langle \{\ldots\},\ldots\rangle^*$	analogous to case above

Figure 4. Termination of proof reduction

where s is S-reducible at a position which superposes at p. We will see later that there might be several ways to reduce them, hence we give for the moment a generic definition for doing that: *normalizing pairs.*

DEFINITION 3.5. *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-*normalizing pair *(w.r.t. $\succ_{\mathcal{P}}$) if for any terms u and v such that $u \succ v$:*

(i) for any elementary irreducible proof of the form

$$s \underset{u=v}{\longleftrightarrow} t$$

there exists a smaller proof (w.r.t. $\succ_{\mathcal{P}}$) in $\Theta(u, v) \cup \Psi(u, v)$ between s and t;

(ii) for all $l \to r \in \Psi(u, v)$, for all sets of rules R, for all r' such that $r \to^*_{R/S} r'$, for any elementary irreducible proof of the form

$$s \underset{l \to r'}{\longrightarrow} t$$

there exists a smaller proof (w.r.t. $\succ_{\mathcal{P}}$) in $\Theta(u, v) \cup \Psi(u, v) \cup R$ between s and t;

The role of the two previous properties will become clear when we will give the proof reduction rules reflecting the completion process.

3.2. Inference rules for normalized completion

As now customary, we describe the completion process by a set of inference rules (Figure 5). $\succeq$ is a reduction ordering, E is a set of equations and R is a set of rules. In the rule COLLAPSE, the ordering $\succ\!\!\succ$ is the lexicographic combination of $\trianglerighteq$ and $\succ$, that is $(u, v) \succ\!\!\succ (l, r\theta)$ means $p \neq \Lambda$, or $p = \Lambda$ and θ is not a renaming, or $p = \Lambda$, θ renaming and $u \succ r\theta$.

Notice that the rule ORIENT does not act anymore as usual by simply turning an equation into a rule, it rather transforms an equation into its normalizing pair.

The rule DEDUCE computes critical pairs modulo some equational theory T that we can choose arbitrarily between AC and S, the validity of this fact being justified by the proof reduction rule we will consider (see second remark after the proof of the completeness theorem).

This is a very important point for two reasons:

(i) S may not be decidable and finitary with respect to unification. For example, S may contain distributivity law, and we know that unification modulo ACD is undecidable (Kirchner, 1990). In such a case we should use $T = \text{AC}$.

(ii) It is known that ACU-unification and ACUI-unification lead to complete sets of unifiers which are usually much smaller than AC unification (Bürckert et al., 1988; Domenjoud, 1992; Kirchner, 1990). A well-known example is the equation $x+x+x = y+z+t+u$ which have 1044569 most general AC solutions but only one most general ACU-solution.

DEFINITION 3.6. *An S-*normalized completion algorithm *is an algorithm which takes as*

ORIENT

$$E \cup \{u = v\}; R \vdash E \cup \Theta(u, v); R \cup \Psi(u, v) \quad \text{if } u = u\downarrow_S,\ v = v\downarrow_S,\ u \succ v.$$

DEDUCE

$$E; R \vdash E \cup \{u = v\}; R \quad \text{if } u = v \in CP_T(R).$$

NORMALIZE

$$E \cup \{u = v\}; R \vdash E \cup \{u\downarrow_S = v\downarrow_S\}; R$$

DELETE

$$E \cup \{u = v\}; R \vdash E; R \quad \text{if } u =_{\mathrm{AC}} v.$$

COMPOSE

$$E; R \cup \{u \to v\} \vdash E; R \cup \{u \to v'\} \quad \text{if } v \xrightarrow[R/S]{} v'.$$

SIMPLIFY

$$E \cup \{u = v\}; R \vdash E \cup \{u' = v\}; R \quad \text{if } u \xrightarrow[R/S]{} u'.$$

COLLAPSE

$$E; R \cup \{u \to v\} \vdash E \cup \{u' = v\}; R \quad \text{if } l \to r \in R,\ u \xrightarrow[l \to r/S]{\theta, p} u',\ (u, v) \succ (l, r\theta).$$

Figure 5. Inference rules of normalized completion

input a set of equations E_0 and a reduction ordering $\succeq$ 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:

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

E_∞ and R_∞ are respectively the set 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.

DEFINITION 3.7. *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 worthwhile 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.

THEOREM 3.8. *Assume we have an S-normalizing pair (Θ, Ψ) (w.r.t. $\succ_{\mathcal{P}}$). Assume that the completion is fair and succeeds. Then for all s and t, $s =_{E_0 \cup AC \cup S} t$ if and only if*

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

PROOF. This result is proved by the proof normalization method (Bachmair et al., 1986; Bachmair, 1991; Bachmair and Dershowitz, 1989; Marché, 1993). Let us consider two terms s and t such that $s =_{E_0 \cup AC \cup S} t$. We have then an equational proof P_0 of that equality. Our aim is to transform this proof into a proof of the desired form. For that, we reduce this proof P_0 by the relation $\Longrightarrow$. To reflect the effect of completion inference rules, we have to add new rules, given in Figure 6.

Let us comment on the rules ORIENT and REORIENT, which correspond to the two properties characterizing normalizing pairs: ORIENT will turn an equational step which possesses a critical superposition with S into a proof in $\Theta \cup \Psi$ according to property (i) in the definition of normalizing pairs. REORIENT is needed to be able to do the same with rewriting steps: one can remark that in the inference rules of the completion, we never compute the pairs (Θ, Ψ) associated to rules, only to equations. For that reason, to be able to reduce rewriting steps where a critical superposition with S occurs, we want to use the pair $(\Theta(u, v), \Psi(u, v))$, $u = v$ being the equation from which the rule comes. But a rule appearing in the completion may come either from a set $\Psi(u, v)$, or by composition of another rule: that explains the role of the set of rules R in the property (ii) in the definition of normalizing pairs and in the rule REORIENT.

Let us denote $P \Longrightarrow_{E;R} Q$ if P and Q are $E; R$-proofs and $P \Longrightarrow Q$, and:

$$\mathcal{E}_i = \bigcup_{j=0}^{i} E_j \text{ and } \mathcal{R}_i = \bigcup_{j=0}^{i} R_j.$$

First of all, if $E; R \vdash E'; R'$ and P is an $E; R$-proof then $P \Longrightarrow^*_{E \cup E'; R \cup R'} Q$ where Q is an $E'; R'$-proof: this is a simple but technical verification that rules of Figure 6 reflect correctly the inferences.

Secondly, for all proofs P and Q, $P \Longrightarrow Q$ implies $P \succ_{\mathcal{P}} Q$: Verification of decreasingness of the reductions reflecting inferences is made in Figure 7. One has to remark that in several cases of a rule $P \Longrightarrow Q$ where $C(Q)$ contains a $c(s, p, t)$ but without knowing we are in the case $s = s\downarrow_p$ or not, we make the proof with $\{s\downarrow_p, t\downarrow_p\}$, which is always greater or equal to $c(s, p, t)$.

Now we know that reduction of proofs terminates, let us denote by $P\downarrow_i$ the normal form of P with respect to $\Longrightarrow_{\mathcal{E}_i;\mathcal{R}_i}$. Now, we remark that P_i is actually an $E_i; R_i$-proof: if it uses an equation in $\mathcal{E}_i$ which is not in E_i then the inference which has removed this equation can be reflected to a reduction of P, hence P is not in normal form (and analogously if it uses a rule in $\mathcal{R}_i$ not in R_i).

Now, let us consider the sequence of proofs starting from P_0 defined by $P_{i+1} = P_i\downarrow_{i+1}$. Since $\Longrightarrow$ is well-founded, this sequence cannot decrease infinitely, there exists a n such

ORIENT

$$s \xleftrightarrow[u=v]{\sigma,p} t \Longrightarrow \text{a proof in } \Theta(u,v) \cup \Psi(u,v) \quad \text{if } u \succ v.$$

REORIENT

$$s \xrightarrow[l \to r']{\sigma,p} t \Longrightarrow \text{a proof in } \Theta(u,v) \cup \Psi(u,v) \cup R$$
$$\text{with } l \to r \in \Psi(u,v),\ r \xrightarrow[R/S]{*} r'.$$

DEDUCE

$$s \xleftarrow[l_1 \to r_1]{\sigma,p} t \xrightarrow[l_2 \to r_2]{\sigma,pq} u \Longrightarrow s \xleftrightarrow[S]{(\geq p)^*} s' \xleftrightarrow[u=v]{\sigma',p} u' \xleftrightarrow[S]{(\geq p)^*} u$$
$$\text{with } \theta \in \mathrm{CSU}_T(l_1|_q, l_2) \mid \sigma \equiv \theta\sigma',\ u = v \in CP_T(l_1 \to r_1, l_2 \to r_2).$$

DELETE

$$s \xleftrightarrow[l=r]{\sigma,p} t \Longrightarrow s \xleftrightarrow[\mathrm{AC}]{*} t \quad \text{if } l \equiv r.$$

COMPOSE

$$s \xrightarrow[u \to v]{\sigma,p} t \Longrightarrow s \xrightarrow[u \to v']{\sigma,p} s' \xleftarrow[l \to r]{\theta\sigma,pq} t' \xleftrightarrow[\mathrm{AC}]{*} t \quad \text{if } v \xrightarrow[l \to r/S]{\theta,q} v' \text{ and } s = s{\downarrow_S}.$$

SIMPLIFY

$$s \xleftrightarrow[u=v]{\sigma,p} t \Longrightarrow s \xrightarrow[l \to r]{\theta\sigma,pq} t' \xleftrightarrow[u'=v]{\sigma,p} t \quad \text{if } u \xrightarrow[l \to r/S]{\theta,q} u' \text{ and } s = s{\downarrow_S}.$$

COLLAPSE

$$s \xrightarrow[u \to v]{\sigma,p} t \Longrightarrow s \xrightarrow[l \to r]{\theta\sigma,pq} t' \xleftrightarrow[u'=v]{\sigma,p} t \quad \text{if } u \xrightarrow[l \to r/S]{\theta,q} u',\ (u,v) \succ (l, r\theta) \text{ and } s = s{\downarrow_S}.$$

Figure 6. Proof rewriting: rules reflecting inference rules

that $P_n = P_k$ for all $k \geq n$. By the previous remark, P_n is an $E_k; R_k$-proof for all $k \geq n$, hence it is an $E_\infty; R_\infty$-proof. Since we assume that completion does not fail, E_∞ is empty hence P_n does not contain any equational steps.

We are left to prove that P_n does not contain any peak pattern: by contradiction, if P_n contains a peak pattern, then first it is necessarily a critic one otherwise a commutation rule applies, and second, by the fairness property, the critical pairs associated to this peak have necessarily been computed at some step j of the completion, but then P_n is not in normal form with respect to $\Longrightarrow_{\mathcal{E}_i;\mathcal{R}_i}$ where $i = \max(n,j)$, a contradiction.

As a consequence P_n is of the form

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

The only if part of the theorem is trivial, since each inference preserves the underlying equational theory. □

rule $P \Longrightarrow Q$	$C(P)$	$C(Q)$	proof of $P \succ_P Q$
ORIENT			By definition of normalizing pairs
DEDUCE	$\langle\{t\},\ldots\rangle$ $\langle\{t'\},\ldots\rangle$	$\langle\bot,\ldots\rangle^*$ $\langle\{s'{\downarrow_P}, u'{\downarrow_P}\},\ldots\rangle$ $\langle\bot,\ldots\rangle^*$	$t = t{\downarrow_S}$, $t \succ s \succeq s'{\downarrow_P}$ and $t' \succ u \succeq u'{\downarrow_P}$
DELETE	$\langle\{s{\downarrow_P}, t{\downarrow_P}\},\ldots\rangle$	$\langle\bot,\ldots\rangle$	trivial
COMPOSE	$\langle\{s\},\{s\},u\sigma,v\sigma\rangle$	$\langle\{s\},\{s\},u\sigma,v'\sigma\rangle$ $\langle c(t',pq,s'),\ldots\rangle$ $\langle\bot,\ldots\rangle$	$v \succ v'$, $c(t'pq,s') = \{t'\}$ or $\{t',s'\}$ and $s \succ t \equiv t' \succ s'$
SIMPLIFY	$\langle\{s{\downarrow_P}, t{\downarrow_P}\},\ldots\rangle$	$\langle\{s\},\ldots\rangle$ $\langle\{t'{\downarrow_P}, t{\downarrow_P}\},\ldots\rangle$	$s = s{\downarrow_S}$ and $s \succ t' \succeq t'{\downarrow_P}$
COLLAPSE	$\langle\{s\},\{s\},u\sigma,v\sigma\rangle$	$\langle\{s\},\{s\},l\theta\sigma,r\theta\sigma\rangle$ $\langle\{t'{\downarrow_P}, t{\downarrow_P}\},\ldots\rangle$	$s = s{\downarrow_S}$, $(u,v) \succ (l, r\theta)$, $s \succ t'$ and $s \succ t$

Figure 7. Termination of proof reduction (rules reflecting completion inferences)

Let us end this section by two remarks:

(i) Since the definition of normalizing pairs is generic in some sense, it is not satisfactory that the ordering itself is not generic. Of course things could also have been made generic w.r.t. the ordering, but it would make less clear the presentation, and in any case we do not know any other ordering that makes decrease all proof reduction rules (In fact, the search for such an ordering is certainly the harder part of the completeness theorem).

(ii) Let us discuss about the use of T-unification in the rule DEDUCE. We want to reduce critical peaks of the form

$$s \xleftarrow[l_1 \to r_1]{\sigma,p} t \xrightarrow[l_2 \to r_2]{\sigma,pq} u$$

We know that $l_1|_p \equiv l_2$ (or with additional extension variables, see Section 2.2) hence we are able to reduce this peak via an AC-critical pair:

$$s \xleftrightarrow[AC]{(\geq p)^*} s' \xleftrightarrow[u=v]{\sigma',p} u' \xleftrightarrow[AC]{(\geq p)^*} u$$

but we know that the equational theory we are completing contains S, hence if T is a sub-theory of S which behaves well with respect to unification (such as ACU), Make rewrite step

$$s \xleftrightarrow[u=v]{\sigma,p} t \Longrightarrow s \xrightarrow[u\to v]{\sigma,p} t \quad \text{if } s = s{\downarrow_S} \text{ and } u \succ v.$$

Critical instance of an equation

$$s \xleftrightarrow[u=v]{\sigma,p} t \Longrightarrow s \xrightarrow[l\to r]{\sigma,q}_S s' \xleftrightarrow[\mathrm{AC}]{*} s'' \xleftrightarrow[u\theta[r\theta]_{q'}=v\theta]{\lambda,p} t' \xleftrightarrow[\mathrm{AC}]{*} t$$

if $u \succ v$, $q = pq'$ with $q' \in \mathcal{FP}\mathrm{os}(u)$, σ S-irreducible, $u|_{q'}\sigma \equiv s|_p|_{q'}\sigma \equiv s|_q\sigma \equiv l\sigma$ hence $\exists\theta \in \mathrm{CSU}_{\mathrm{AC}}(u|_{q'}, l)\exists\lambda \mid \sigma \equiv \theta\lambda$.

Critical contextual equation of an equation

$$s \xleftrightarrow[u=v]{\sigma,p} t \Longrightarrow s \xrightarrow[l\to r]{\sigma,q}_S s' \xleftrightarrow[\mathrm{AC}]{*} s'' \xleftrightarrow[r\theta=l\theta[v\theta]_{p'}]{\lambda,q} t' \xleftrightarrow[\mathrm{AC}]{*} t$$

if $s \succ t$, $s|_p$ S-irreducible, $p = qp'$ with $p' \in \mathcal{FP}\mathrm{os}(l)$ and $p' \neq \Lambda$, σ S-irreducible, $l|_{p'}\sigma \equiv s|_{qp'}\sigma \equiv u\sigma$ hence $\exists\theta \in \mathrm{CSU}_{\mathrm{AC}}(l|_{p'}, u)\exists\lambda \mid \sigma \equiv \theta\lambda$.

Critical instance of a rule

$$s \xrightarrow[u\to v']{\sigma,p} t \Longrightarrow s \xrightarrow[l\to r]{\sigma,q}_S s' \xleftrightarrow[\mathrm{AC}]{*} s'' \xleftrightarrow[u\theta[r\theta]_{q'}=v\theta]{\lambda,p} t' \xrightarrow[R/S]{*} t$$

if $v \xrightarrow[R/S]{*} v'$, $q = pq'$ with $q' \in \mathcal{FP}\mathrm{os}(u)$, σ S-irreducible, $u|_{q'}\sigma \equiv s|_p|_{q'}\sigma \equiv s|_q\sigma \equiv l\sigma$ hence $\exists\theta \in \mathrm{CSU}_{\mathrm{AC}}(u|_{q'}, l)\exists\lambda \mid \sigma \equiv \theta\lambda$.

Critical contextual equation of a rule

$$s \xrightarrow[u\to v']{\sigma,p} t \Longrightarrow s \xrightarrow[l\to r]{\sigma,q}_S s' \xleftrightarrow[\mathrm{AC}]{*} s'' \xleftrightarrow[r\theta=l\theta[v\theta]_{p'}]{\lambda,q} t' \xrightarrow[R/S]{*} t$$

if $v \xrightarrow[R/S]{*} v'$, $p = qp'$ with $p' \in \mathcal{FP}\mathrm{os}(l)$ and $p' \neq \Lambda$, σ S-irreducible, $l|_{p'}\sigma \equiv s|_{qp'}\sigma \equiv u\sigma$ hence $\exists\theta \in \mathrm{CSU}_{\mathrm{AC}}(l|_{p'}, u)\exists\lambda \mid \sigma \equiv \theta\lambda$.

Figure 8. Proof rewriting: general S-normalizing pair

we can reduce the peak via a T-critical pair:

$$s \xleftrightarrow[S]{(\geq p)^*} s' \xleftrightarrow[u=v]{\sigma',p} u' \xleftrightarrow[S]{(\geq p)^*} u$$

the S-steps above are in fact T-steps (we use S because T-steps are not in the proof algebra), needed because since we use T-unification, σ is a T-instance of a most general T-unifier, not necessarily an AC-instance.

Moreover, this shows that in the computation of CP_T, we can restrict ourselves to T-unifiers *which have at least one AC-unifier as an instance.* This remark can be used to reduce a bit more the set critical pairs computed in practice: for example, $x + y$ and a have two ACU-unifiers $x \mapsto a, y \mapsto 0$ and $x \mapsto 0, y \mapsto a$ but none have AC-unifiers as instances. We will use this also in the next section.

rule $P \Longrightarrow Q$	$C(P)$	$C(Q)$	proof of $P \succ_P Q$
MAKE REWRITE STEP	$\langle\{s,t\},\ldots\rangle$	$\langle c(s,p,t),\ldots\rangle$	$s = s{\downarrow_S}$ hence $c(s,p,t) = \{s\} \prec \{s,t\}$
CRITICAL INSTANCE OF AN EQUATION	$\langle\{s{\downarrow_P}, t{\downarrow_P}\}$, $\{s,t\},\ldots\rangle$	$\langle\{s''{\downarrow_P}, t'{\downarrow_P}\}$, $\{s'',t'\}\ldots\rangle$ $\langle\bot,\ldots\rangle^*$	$s{\downarrow_P}= s''{\downarrow_P}$, $t{\downarrow_P}= t'{\downarrow_P}$, $s \succ s' \equiv s''$ and $s \succ t \equiv t'$
CRITICAL CONTEXTUAL EQUATION OF AN EQUATION	$\langle\{s{\downarrow_P}, t{\downarrow_P}\}$, $\{s,t\},\ldots\rangle$	$\langle\{s''{\downarrow_P}, t'{\downarrow_P}\}$, $\{s'',t'\}\ldots\rangle$ $\langle\bot,\ldots\rangle^*$	same as above.
CRITICAL INSTANCE OF A RULE	$\langle\{c(s,p,t),\ldots\rangle$	$\langle\{s''{\downarrow_P}, t'{\downarrow_P}\},\ldots\rangle$ $\langle c(t_i,p_i,t_{i+1}),\ldots\rangle$ $\langle\bot,\ldots\rangle^*$	$c(s,p,t) \succeq_{\mathrm{mul}} \{s\}$, $s \succ s' \equiv s'' \succeq s''{\downarrow_P}$; $s = s[u\sigma]_q$, $t' = s[v\sigma]_q$ and $u \succ v$ hence $s \succ t'$; $t' \succeq t_i$ for all i.
CRITICAL CONTEXTUAL EQUATION OF A RULE	$\langle\{c(s,p,t),\ldots\rangle$	$\langle\{s''{\downarrow_P}, t'{\downarrow_P}\},\ldots\rangle$ $\langle c(t_i,p_i,t_{i+1}),\ldots\rangle$ $\langle\bot,\ldots\rangle^*$	same as above.

Figure 9. Termination of proof reduction for the general normalizing pair

3.4. A GENERAL S-NORMALIZING PAIR

We show in this subsection how one can define an S-normalizing pair for an arbitrary canonical rewrite system S.

DEFINITION 3.9. *Let u and v be two terms in S-normal form such that $u \succ v$. Let $\Theta_{\mathrm{gen}}(u,v)$ be the set of equations (called* critical instances*) of the form $u\theta[r\theta]_q = v\theta$ where $q \in \mathcal{FP}\mathrm{os}(u)$, $l \to r \in S$, $\theta \in \mathrm{CSU}_{\mathrm{AC}}(u|_q, l)$ S-irreducible, and the equations (called* critical contextual equations*) of the form $l\theta[v\theta]_q = r\theta$ where $q \in \mathcal{FP}\mathrm{os}(l)$, $q \neq \Lambda$, $l \to r \in S$, $\theta \in \mathrm{CSU}_{\mathrm{AC}}(u, l|_q)$ S-irreducible; and $\Psi_{\mathrm{gen}}(u,v)$ be $\{u \to v\}$.*

PROPOSITION 3.10. *The pair $(\Theta_{\mathrm{gen}}, \Psi_{\mathrm{gen}})$ is S-normalizing with respect to the proof ordering defined above, for any AC-convergent set of rules S.*

PROOF. The proof reduction rules associated to these normalizing pairs are shown in Figure 8: the first one allows to reduce equational steps already in S normal form, the second and third ones allow to reduce equational steps where a critical superposition with S occurs, hence those three rules make property (i) of the definition of normalizing pairs satisfied. The two last rules allow to reduce rewrite steps hence make satisfied property (ii). Proof of decreasingness of these rules is made in Figure 9. □

Notice this definition is not satisfactory since we use AC-unification and in fact $\Theta_{\text{gen}}(u,v)$ simply contains all critical pairs between $u \to v$ and S, hence S-normalized completion will behave as AC-completion with a particular strategy. In the next section, we will try to enhance this definition to obtain a more efficient completion algorithm.

EXAMPLE 3.11. *Assume $S = \{z + 0 \to z\}$ where $+$ is AC. Let us compute $\Theta_{\text{gen}}(-(x + y), (-x) + (-y))$: we have to unify modulo AC the terms $x + y$ and $z + 0$. This leads to 4 most general unifiers:*

$$\begin{cases} x \mapsto v_1 \\ y \mapsto 0 \\ z \mapsto v_1 \end{cases} \begin{cases} x \mapsto 0 \\ y \mapsto v_1 \\ z \mapsto v_1 \end{cases} \begin{cases} x \mapsto v_1 \\ y \mapsto v_2 + 0 \\ z \mapsto v_1 + v_2 \end{cases} \begin{cases} x \mapsto v_1 + 0 \\ y \mapsto v_2 \\ z \mapsto v_1 + v_2 \end{cases}$$

The last two are S-reducible so we ignore them. Hence $\Theta_{\text{gen}}(-(x + y), (-x) + (-y))$ 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 ACU-constrained completion (Jouannaud and Marché, 1992).

3.5. A SIMPLE MODULARITY RESULT

Here is a simple result that will allow us for example to complete commutative ring theory modulo $\mathrm{AG}(+, 0, -) \cup \mathrm{ACU}(*, 1)$.

PROPOSITION 3.12. *Assume S_1 and S_2 are two convergent systems over disjoint signatures and included in the same reduction ordering, then we have*

(i) $S_1 \cup S_2$ is convergent;
(ii) if (Θ_i, Ψ_i) is S_i-normalizing ($i = 1, 2$) then $(\Theta_1 \cup \Theta_2, \Psi_1 \cup \Psi_2)$ is $S_1 \cup S_2$-normalizing.

PROOF. Part (i) is straightforward since a rule of S_1 and a rule of S_2 can never superpose together. Part (ii) is true because whenever we have a proof $s \longleftrightarrow t$ where a critical superposition with $S_1 \cup S_2$ occurs, then it is either a critical superposition with S_1 or with S_2, and we can reduce the proof either into a proof in $\Theta_1 \cup \Psi_1$ or in $\Theta_2 \cup \Psi_2$. □

4. Optimized normalizing pairs

4.1. EXPLOITING LEFT-LINEARITY

When using normalized rewriting modulo a fixed S, we can optimize the definition of the general normalizing pair. In particular, for the rules of S which are left-linear we can avoid the use of AC-unification. The fact that general E-unification can be avoided for left-linear rules (if E congruence classes are finite) has been first mentioned by Huet (Huet, 1980): to compute E-critical pairs between to left-linear rules, one only needs to compute standard critical pairs with all E-variants of rules.

4.1.1. SOME SIMPLE THEORIES

Figure 10 shows definitions of Θ_S where S is either ACU, ACI, ACUI, AC0 or ACN, and

theory S	convergent system	$\Theta_S(u,v)$
ACU(+,0)	$x+0 \to x$	$\{u\theta = v\theta \mid \theta = x \mapsto 0, u \trianglerighteq x+w\}$
ACI(+)	$x+x \to x$	$\mathrm{CP}_{\mathrm{AC}}(u \to v, x+x \to x)$
ACUI(+,0)	$x+x \to x$, $x+0 \to x$	$\Theta_{\mathrm{ACU}}(u,v) \cup \Theta_{\mathrm{ACI}}(u,v)$
AC0(∗,0)	$x*0 \to 0$	$\{u\theta = v\theta \mid \theta = x \mapsto 0, u \trianglerighteq x*w\}$
ACN(+,0)	$x+x \to 0$	$\mathrm{CP}_{\mathrm{AC}}(u \to v, x+x \to 0)$

Figure 10. Set Θ_S for some simple theories

in all these cases $\Psi_S(u,v) = \{u \to v\}$. notice also that in the cases ACI and ACN $\mathrm{CP}_{\mathrm{AC}}$ is the AC-critical pairs of the second rule *inside* the first: the inverse is not necessary since rules defining ACI and ACN have depth 1. Also, remember that $\mathrm{CP}_{\mathrm{AC}}$ uses the generalized notion of overlapping defined in Section

4.2. SUBSECTION:REWRITING-ON-FLAT-TERMS

, that is we also have to compute overlappings with the extensions $x+x+y \to x+y$ and $x+x \to y$.

PROPOSITION 4.1. *If S is either ACU, ACI, ACUI, AC0 or ACN, the above defined mappings Θ_S, Ψ_S are S-normalizing.*

PROOF. Straightforward: Θ_S and Ψ_S are in these cases simply direct computations of Θ_{gen} and Ψ_{gen}. □

4.2.1. Associativity

Another interesting case is when S is the theory of associativity of one symbol (in that case we don't have AC symbols anymore, but of course what we have done before is still valid!). For the canonical rewrite system S we can choose an orientation for associativity (by use of a lexicographic path ordering for example), let's say for example $S = \{(x*y)*z \to x*(y*z)\}$.

In that case we can avoid many useless critical pairs. The idea is that deduction between S and a rule of the form $s_1 * \cdots * s_n \to t$ will only produce one useful deduction, which can be seen as an A-extension of the original rule: $s_1 * \cdots * s_n * x \to t * x$ where x is a new variable. We have of course also to compute an optimized set Θ: it can be done avoiding call to unification by simply looking for subterms of the form $x * s$ where x is a variable.

DEFINITION 4.2. *Let $\Psi_{\mathrm{A}}(u,v) = \{u \to v\}$ if $\mathcal{H}\mathrm{ead}(u) \neq *$ and*

$$\Psi_{\mathrm{A}}(u_1 * \cdots * u_n, v) = \begin{cases} u_1 * \cdots * u_n \to v \\ u_1 * \cdots * u_n * x \to v * x \end{cases}$$

otherwise. Let

$$\Theta_{\mathrm{A}}(u,v) = \{u\theta = v\theta \mid \theta = x \mapsto x*y, u \trianglerighteq x*w\}$$

PROPOSITION 4.3. *The above defined mapping* $(\Theta_{\mathrm{A}},\ \Psi_{\mathrm{A}})$ *is* A*-normalizing.*

PROOF. The part $\Theta_{\mathrm{gen}}(u,v)$ where superpositions of $u \rightarrow v$ inside rules of S are computed yields the additional rule in Ψ_{A}, whereas the part where superpositions of S inside $u \rightarrow v$ are computed yields to Θ_{A}. □

EXAMPLE 4.4. *We can complete group theory modulo A by giving to the A-normalized completion the set of equations*

$$\left\{ \begin{array}{ll} x * e = x & e * x = x \\ x * I(x) = e & I(x) * x = e \end{array} \right.$$

Unlike Knuth-Bendix completion, the new rule $x * (I(x) * y) \rightarrow y$ and the new equation $x * (y * I(x * y)) = e$ will be obtained without calling unification, because they are in $\Psi_{\mathrm{A}}(I(x) * x, e)$ and $\Theta_{\mathrm{A}}(I(x) * x, e)$ respectively.

4.3. Using symmetrization

When S contains at least Abelian group theory, we can optimize much further the normalizing pair by using *symmetrization*. 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 of *symmetrization* is inspired by (Le Chenadec, 1986). We use the abbreviation nt for $\underbrace{t + \cdots + t}_{n \text{ times}}$.

DEFINITION 4.5. *The* symmetrization *of a pair (u,v) is obtained in the following way: let w be the AG-normal form of $u + (-v)$, written as $w = n_1 w_1 + \cdots + n_k w_k$, with $\forall j \geq 2$, $w_1 \succ w_j$. Then $\mathrm{sym}(u,v) = (n_1, w_1, -n_2 w_2 - \cdots - n_k w_k)$. If there is no maximum w_i, $\mathrm{sym}(u,v)$ is undefined.*

4.3.1. Abelian group theory

DEFINITION 4.6. *For a pair (u,v) that has a symmetrization (n,s,t), let $\Psi_{\mathrm{AG}}(u,v) = s \rightarrow t$ if $n = 1$ and*

$$\Psi_{\mathrm{AG}}(u,v) = \left\{ \begin{array}{l} ns \rightarrow t \\ -s \rightarrow (n-1)s + (-t) \end{array} \right.$$

if $n \geq 2$ (notice that the right-side of the second rule may need further AG-normalization). Let

$$\Theta_{\mathrm{AG}}(u,v) = \Theta_{\mathrm{ACU}}(ns,t) \cup \Sigma_1(ns,t) \cup \Sigma_2(ns,t)$$

where

$$\Sigma_1(u,v) = \mathrm{CP}_{\mathrm{AC}}(u \rightarrow v, x + (-x) \rightarrow 0)$$

where superpositions are computed only in strict subterms of u, and

$$\Sigma_2(u,v) = \{u\sigma = v\sigma \mid \sigma = x \mapsto 0 \textit{ or } -y \textit{ or } y + z \textit{ if } u \trianglerighteq -x\}$$

If (u,v) does not have a symmetrization, the equation $u = v$ will be considered as not orientable.

SYMMETRIZATION 1

$$s \xleftrightarrow[u+v=w]{\sigma,p} t \Longrightarrow s \xleftrightarrow[u=(-v)+w]{\sigma,p'} t' \xrightarrow[S]{(\geq p)*} t \quad \text{if } s = s{\downarrow_S},\ u \succ v \text{ and } u \succ w.$$

SYMMETRIZATION 2

$$s \xleftrightarrow[u+(-v)=w]{\sigma,p} t \Longrightarrow s \xleftrightarrow[u=v+w]{\sigma,p'} t' \xrightarrow[S]{(\geq p)*} t \quad \text{if } s = s{\downarrow_S},\ u \succ v \text{ and } u \succ w.$$

SYMMETRIZATION 3

$$s \xleftrightarrow[(-u)+(-u)+v=w]{\sigma,p} t \Longrightarrow s \xleftrightarrow[(-u)+v=u+w]{\sigma,p'} t' \xrightarrow[S]{(\geq p)*} t \quad \text{if } s = s{\downarrow_S},\ u \succ v.$$

CANCELATION

$$s \xleftrightarrow[u+w=v+w]{\sigma,p} t \Longrightarrow s \xleftrightarrow[u=v]{\sigma',p'} t \quad \text{if } s = s{\downarrow_S}.$$

Figure 11. Proof reduction: symmetrization

EXAMPLE 4.7. *If $u = a + a + (-b) + c + c$ and $v = a + b + b + c$, then $u + (-v)$ normalizes to $a + (-b) + (-b) + (-b) + c$. If the ordering makes a greater than $(-b)$ and c, then $\mathrm{sym}(u, v) = (1, a, b + b + b + (-c))$ and $\Psi_{\mathrm{AG}}(u, v) = \{a \to b + b + b + (-c)\}$. If the ordering makes $-b$ greater than a and c, then $\mathrm{sym}(u, v) = (3, b, (-a) + (-c))$ and $\Psi_{\mathrm{AG}}(u, v) = \{b + b + b \to (-a) + (-c), -b \to b + b + a + c\}$.*

The pair above defined is AG-normalizing only if the ordering $\succ$ satisfies a particular property w.r.t. the operators $+$ and $-$.

PROPOSITION 4.8. *Let us assume that the term ordering satisfies the following property: for all terms u, v and w which do not have $+$, 0 or $-$ at the top, if $u \succ v$ and $u \succ w$ then $u \succ (-v) + w$. Then the pair (Θ_{AG}, Ψ_{AG}) defined above is AG-normalizing.*

PROOF. Symmetrization induces new proof reduction rules shown in Figure 11, and the inference of symmetrization can be reflected by several applications of these rules and the rules for the general S-normalizing pair. These rules are sufficient for reflecting symmetrization at the proof level: let us consider an equational step

$$s \xleftrightarrow[u=v]{\sigma,p} t$$

with $u \succ v$, let us assume $\mathrm{sym}(u, v) = (n, s, t)$. By the CANCELATION rule we can remove common subterms of u and v (hence reflecting at the proof level the AG-normalization of $u + (-v)$). Furthermore, s occurs necessarily in u since $u \succ v$, hence there are two cases:

1 s occurs in u under a minus sign: there are no occurrences of $-s$ in v (because of cancelation), applying SYMMETRIZATION rules will move all subterms of u to the right except one $-s$, hence the equation applied will be $-s \to t$.

2 s does occur in u under a minus sign: there are no occurrences of s in v (because of cancelation), applying the two first SYMMETRIZATION rules will move all subterms of u to the right except all occurrences of s, hence the equation applied will be $ns \to t$.

We remark finally that these rules decrease w.r.t. the proof ordering, because of the property we assumed on the term ordering. □

An ordering satisfying the property above could be for example a precedence-based ordering (like the recursive path ordering if there is only $+$ as AC symbol, or the associative path ordering and its extensions if there are other AC symbols), with a precedence $- > + > 0$ and all other symbols greater than $-$.

This symmetrization technique improves a lot over standard AC completion when the set of equations to complete contains Abelian group theory. Here are some examples.

EXAMPLE 4.9. *During the completion of commutative ring theory modulo AG, the equation* $(x*y)+(x*0) = x*y$ *is generated. The orientation via symmetrization produces the rule* $x*0 \to 0$. *We see in this case that the symmetrization technique includes in particular* cancelation. *Another equation generated during this completion is* $(x*y)+(x*(-y)) = 0$. *Symmetrization gives directly the rule* $x*(-y) \to -(x*y)$, *without computing any AC critical pair, as in the usual AC completion.*

Improvement in practice will be shown in section 6.

It is possible to apply the symmetrization technique to normalized completion modulo commutative ring theory, Boolean ring theory, and also to theories defining finite fields (Marché, 1993). Unfortunately, there is no convergent system for field theory, because of the negative conditional equation $x * x^{-1} = 1$ if $x \neq 0$.

4.3.2. COMMUTATIVE RING THEORY

DEFINITION 4.10. *For a pair* (u, v) *that has a symmetrization* (n, s, t), *let* $\Psi_{\mathrm{CR}}(u, v) = \{s \to t\}$ *if* $n = 1$ *and*

$$\Psi_{\mathrm{CR}}(u,v) = \left\{ \begin{array}{ll} ns \to t & n(x*s) \to x*t \\ -s \to (n-1)s + (-t) & -(x*s) \to (n-1)(x*s) + -(x*t) \end{array} \right.$$

if $n \geq 2$. *Let*

$$\begin{array}{rcl} \Theta_{\mathrm{CR}}(u,v) & = & \Theta_{\mathrm{ACU}(+,0)}(ns,t) \cup \Theta_{\mathrm{ACU}(*,1)}(ns,t) \cup \Theta_{\mathrm{AC0}(*,0)}(ns,t) \cup \\ & & \Sigma_1(ns,t) \cup \Sigma_2(ns,t) \cup \Sigma_3(ns,t) \end{array}$$

where $\Sigma_1(u, v)$ *and* $\Sigma_2(u, v)$ *are the same as for AG, and*

$$\Sigma_3(u,v) = \{u\sigma = v\sigma \mid \sigma = x \mapsto x + y \text{ or } -x \text{ if } u \trianglerighteq x * z\}$$

PROPOSITION 4.11. *If we assume the term ordering satisfying the required property for symmetrization, then the pair* $(\Theta_{\mathrm{CR}}, \Psi_{\mathrm{CR}})$ *defined above is* CR*-normalizing.*

DISTRIBUTION

$$s \xleftrightarrow[u+\cdots+u=v]{\sigma,p.q} t \Longrightarrow s \xrightarrow[S]{*} s' \xleftrightarrow[x*u+\cdots+x*u=x*v]{\sigma',p} t \quad \text{if } s = s{\downarrow_S},\ s(p) = * \text{ and } u \succ v.$$

NEGATION

$$s \xleftrightarrow[-u=v]{\sigma,p.q} t \Longrightarrow s \xrightarrow[S]{*} s' \xleftrightarrow[-(x*u)=x*v]{\sigma',p} t \quad \text{if } s = s{\downarrow_S},\ s(p) = * \text{ and } u \succ v.$$

Figure 12. Proof reduction: distribution and negation

PROOF. $\Psi_{\mathrm{CR}}(u,v)$ contains critical superpositions of $u \to v$ inside $x*(y+z) \to x*y+x*z$ and $x*(-y) \to -(x*y)$, and this induce new proof reduction rules as shown in Figure 12. $\Theta_{\mathrm{CR}}(u,v)$ contains critical superpositions of S inside $u \to v$. □

This optimized normalizing pair allows to compute Gröbner bases much more efficiently than AC-completion, as we will see in Section 6.

4.3.3. BOOLEAN RING THEORY

Boolean ring theory BR is defined by

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

In this case, the definition of normalizing pair is simpler since rules $x + x \to 0$ and $-x \to x$ implies that after a symmetrization, the leading coefficient will always be 1. It is not necessary then to have more than one rule in Ψ_{BR}.

DEFINITION 4.12. *For a pair (u,v) that has a symmetrization $(1,s,t)$, let $\Psi_{\mathrm{BR}}(u,v) = \{s \to t\}$. Let*

$$\begin{array}{rcl} \Theta_{\mathrm{BR}}(u,v) & = & \Theta_{\mathrm{ACU}(+,0)}(s,t) \cup \Theta_{\mathrm{ACU}(*,1)}(s,t) \cup \Theta_{\mathrm{AC0}(*,0)}(s,t) \cup \\ & & \Theta_{\mathrm{ACN}(+)}(s,t) \cup \Sigma_4(ns,t) \end{array}$$

where

$$\Sigma_4(u,v) = \{u\sigma = v\sigma \mid \sigma = x \mapsto x+y \text{ if } u \trianglerighteq x*z\}$$

PROPOSITION 4.13. *The pair $(\Theta_{\mathrm{BR}}, \Psi_{\mathrm{BR}})$ defined above is* BR*-normalizing.*

4.3.4. FINITE FIELDS THEORY

Finite field theory FF(p) for a prime number p is defined by

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

We can use in these case another kind of symmetrization, which will act similarly to taking the inverse of the leading coefficient: if we have an equation of the form $ns = t$ then we would like to symmetrize it into $s \to (n^{-1} \bmod p)t$ in some sense. In the finite field $\mathrm{FF}(p)$ we do not need an inverse operator to do that because $n^{-1} \equiv n^{p-2} \pmod p$. More formally, we know that for any terms s and t, and any positive integer n, the equation $s = \overline{n}t$ (where $\overline{n}$ denotes $n^{p-2} \bmod p$) is an equational consequence of $ns = t$ and $\mathrm{FF}(p)$.

Definition 4.14. *For a pair (u, v) that has a symmetrization (n, s, t), let $\Psi_{\mathrm{FF}}(u, v) = \{s \to \overline{n}t\}$. Let*

$$\begin{aligned}\Theta_{\mathrm{FF}}(u,v) &= \Theta_{\mathrm{ACU}(+,0)}(s,\overline{n}t) \cup \Theta_{\mathrm{ACU}(*,1)}(s,\overline{n}t) \cup \Theta_{\mathrm{AC0}(*,0)}(s,\overline{n}t) \cup \\ &\quad \Sigma_4(s,\overline{n}t) \cup \Sigma_5(s,\overline{n}t)\end{aligned}$$

where $\Sigma_4(u, v)$ is the same as for CR, and

$$\Sigma_5(u,v) = \mathrm{CP}_{\mathrm{AC}}(u \to v, px \to 0)$$

where superpositions are computed only in strict subterms of u.

Proposition 4.15. *The pair $(\Theta_{\mathrm{FF}}, \Psi_{\mathrm{FF}})$ defined above is $\mathrm{FF}(p)$-normalizing.*

We will see in Section 6 an example which shows how to compute Gröbner bases of polynomial ideals over $\mathrm{FF}(p)$ with $\mathrm{FF}(p)$-normalized completion, more efficiently than with AC-completion (it has already been remarked by Bündgen that computation of such a Gröbner basis can be done by AC completion (Bündgen, 1991a)).

5. Decidability of the word problem for some classes of equational theories

Now, we investigate termination issues of the completion process when the initial set of equations is ground. It is already known that ground AC-completion terminates (Narendran and Rusinowitch, 1991; Marché, 1991), and here we extend this result to S-normalized ground completion for some interesting theories S.

5.1. General results

We first look at some general results, true for arbitrary S. To prevent ground completion from failure, we need to assume that the AC ordering we use is *total on ground terms*. It is not very easy to define such an ordering, but it is possible (Narendran and Rusinowitch, 1991; Nieuwenhuis and Rubio, 1993; Marché, 1993).

We define the notion of *generator set* of a term. This extends Narendran and Rusinowitch's definition (Narendran and Rusinowitch, 1991). Let F be the set of functions symbols that appear in S or are AC.

Definition 5.1. *Let u be a (flat) term. The* generator set *of u (w.r.t F) is defined by $\gamma_F(u) = \{\}$ if u is a variable, otherwise $\gamma_F(u) = \{u\}$ if $\mathcal{H}\mathrm{ead}(u) \notin F$ and $\gamma_F(u) = \bigcup_{1 \le i \le n} \gamma_F(u_i)$ if $u = f(u_1, \ldots, u_n)$ with $f \in F$. For a set of equations E and a set of rules R, we denote by $\Gamma_F(E)$ (resp. $\Gamma_F(R)$) the union of generator sets of all members of equations of E (resp. rules of R). Finally, the* generator set *of E and R, denoted $G_F(E, R)$, is the union of $\Gamma_F(E)$ and $\Gamma_F(R)$.*

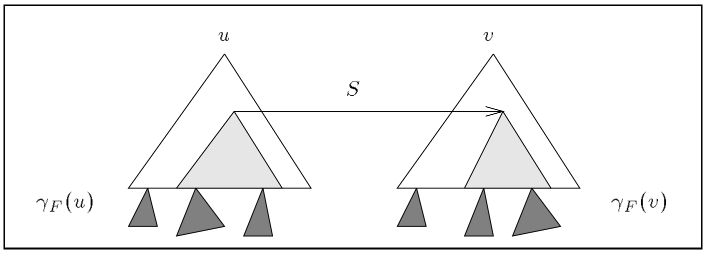

Figure 13. Decreasingness of generator set (case 2.a)

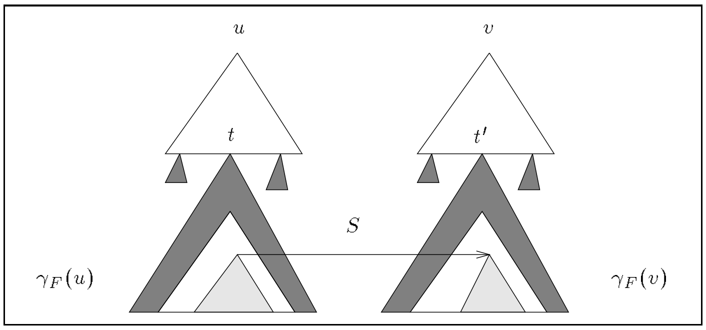

Figure 14. Decreasingness of generator set (case 2.b)

Notice that all these sets are true sets, not multisets. Intuitively, $\gamma_F(u)$ is the set of subterms of u obtained by erasing each top symbol of u which is in F. Note that a symbol of F occurring below a non-F is not erased. For example $\gamma_{+,-,0}(f(a, b+c)+(-d)+0) = \{f(a, b+c), d\}$.

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

Let us first remark that in the case S = CR completion may generate rules with variables even if the initial set is ground. We cannot assume then in the following that the rules we work with are ground, but they still have a particular property: paths between root and variables contain only symbols in F. This is enough to insure that generator sets do not contain variables.

In the following, we consider a set of rules R satisfying this property.

Proposition 5.2. *Assume* $E; R \overset{*}{\vdash} E'; R'$ *is a sequence of simplifications, that is inferences by* Normalize, Delete, Simplify, Compose *or* Collapse, *such that* $E'; R'$ *is no longer simplifyable. Then*

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

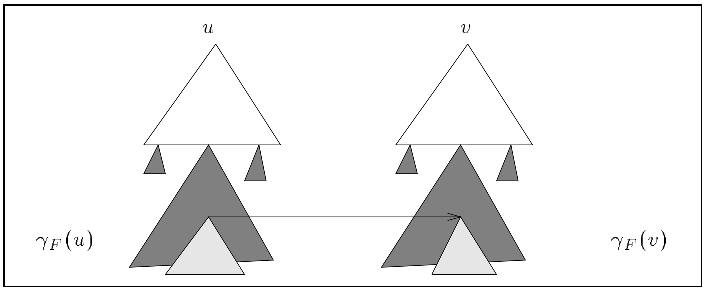

Figure 15. Decreasingness of generator set (case 3.a)

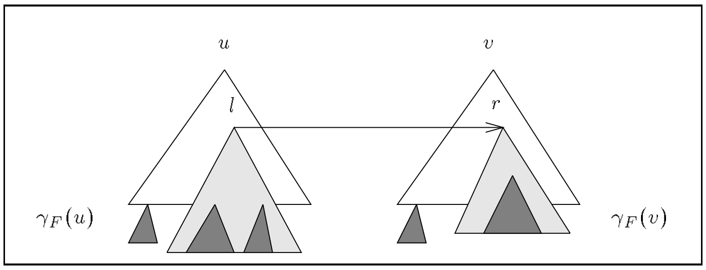

Figure 16. Decreasingness of generator set (case 3.b)

Proof. In the figures mentioned below, the white parts represent symbols in F, the black parts symbols not in F, and gray parts represent redexes.

We reason by induction on the length of the sequence. Let us consider the first step of the derivation $E; R \vdash E''; R'' \overset{*}{\vdash} E'; R'$ (and assume $G_F(E'', R'') \succeq_{\mathrm{mul}} G_F(E', R')$ by induction the hypothesis).

1 If it is Delete: trivial.

2 If it is Normalize: there is a term u in E which is rewritten to v by a rule in S. There are two cases depending whether the position where the rule is applied is inside $\gamma_F(u)$ or not:

(a) if rewriting is done in the top part (Figure 13), the generator set stays the same, or decreases if there rules in S that erase variables (like $x * 0 \to 0$);

(b) otherwise, rewriting is inside a generator subterm t (Figure 14). We cannot be sure that $\gamma_F(u)$ decreases because there may have been several occurrences of t in $E; R$. We can only say that $G_F(E; R) = M \cup \{t\}$ and $G_F(E''; R'') = M \cup \{t, t'\}$. We will conclude at the end of the proof

3 If it is Simplify, Compose or Collapse: there is a term u in E or R which is rewritten to v by a rule $l \to r$ in R.

(a) if there exists $t \in \gamma_F(u)$ such that $u|_p$ is a subterm of t then $\gamma_F(v) = \gamma_F(u) - \{t\} + \{t[r]_q\}$ (where q is the position of t such that $t|_q = u|_p$), we have $t \succ t' =$

$t[r]_q$ (Figure 15), but we have the same remark as before: there may exist several occurrences of t, hence $G_F(E;R) = M \cup \{t\}$ and $G_F(E'';R'') = M \cup \{t,t'\}$.

(b) otherwise, l is built from terms of $\gamma_F(u)$ and symbols in F, in particular $\gamma_F(l) \subseteq \gamma_F(u)$, and $\gamma_F(v)$ contains terms that are already in $\gamma_F(u)$ or in $\gamma_F(r)$ (Figure 16). Notice that this is essential here that rules are ground or that variables do not occur below a symbol not in F, because this implies that $\gamma_F(v)$ really contains terms in $\gamma_F(r)$, not only instances of terms in $\gamma_F(r)$.

We are left to finish the proof of cases 2.b and 3.a: in both cases we have $G_F(E;R) = M \cup \{t\}$, $G_F(E'';R'') = M \cup \{t,t'\}$ where $t \succ t'$. Moreover we know that t is R-simplifyable. This implies that t cannot occur anymore in $G_F(E';R')$ because $E';R'$ is no longer simplifyable and if t is R-simplifyable, it is also R'-simplifyable. (Notice that we don't say that t has to be rewritten to t' because we do not know whether R is confluent, and we do not impose any strategy of application of simplification. It is even possible that for example, an occurrence of t is rewritten by a rule, then this rule being collapsed by another, hence one could imagine that the other occurrences of t could not be rewritten anymore: this is not true because t is still simplifyable, by the rule which collapsed the first rule !)

Let us denote $G_F(E';R')$ by M'. We are now in this situation: $G_F(E;R) = M \cup \{t\}$, $G_F(E'';R'') = M \cup \{t,t'\}$, $t \succ t'$, $t \notin M'$ and $M \cup \{t,t'\} \succeq_{\text{mul}} M'$ by induction hypothesis. By definition of multiset ordering and since $t \notin M'$ we know that $M' = M'' \cup \{t_1,\ldots,t_k\}$ where $M \cup \{t'\} \succeq_{\text{mul}} M''$ and $t \succ t_i$ for all i (k may possibly be 0). But then $M \cup \{t\} \succ_{\text{mul}} M \cup \{t',t_1,\ldots,t_k\}$ since $t \succ t'$, hence $M \cup \{t\} \succ_{\text{mul}} M'' \cup \{t_1,\ldots,t_k\}$ since $M \cup \{t'\} \succeq_{\text{mul}} M''$, hence $M \cup \{t\} \succ_{\text{mul}} M'$ by definition of M''. Hence we have finally obtained $G_F(E;R) \succeq_{\text{mul}} G_F(E';R')$. □

DEFINITION 5.3. *We say that the strategy* simplifies first *if the simplification rules* NORMALIZE, 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 (Marché, 1991). Now let us show the main result of this section.

THEOREM 5.4. *Let us assume the normalizing pair used in completion keeps the same or decreases G_F when rule* ORIENT *or* DEDUCE *is applied. Assume that the 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.*

PROOF. By contradiction: if R_∞ is finite, there exists a j such that $R_j = R_\infty$, and then all the following derivation steps produce equations that are simplified and then deleted (because they are joinable by R_∞) before being eventually used for deduction. Hence derivation as to be finite since $CP_T(R_\infty)$ is finite itself.

Now, let us assume there are infinitely many rules $l \to r$ such that $\mathcal{H}\text{ead}(l) \notin F$. Derivation is of the form

$$E_0;\emptyset \overset{*}{\vdash} E_{n_1};R_{n_1} \overset{+}{\vdash} E_{n_2};R_{n_2} \overset{+}{\vdash} \cdots$$

where all $E_{n_i};R_{n_i}$ are inter-reduced. By proposition 5.2 the sequence of generator sets $G_F(E_{n_i},R_{n_i})$ is decreasing with respect to the well-founded ordering $\succeq_{mul}$, hence it eventually becomes constant. Let us call G_F^∞ this constant value. The left-hand sides of rules of R_∞ whose head symbol is not in F are in the finite set G_F^∞, hence there are some rules that have the same left-hand side. But this is not possible if $\succ$ is total because COLLAPSE can then be applied. □

Let us now discuss the hypothesis that neither ORIENT nor DEDUCE increase the generator set. For ORIENT, this depends upon the choice of the normalizing pair, but the fact is that the hypothesis is true for each choice of normalizing pair we considered earlier: the reason is that in the case of ground completion, the Θ set is always empty (unification of a rule of S and a ground rule produces only joinable critical pairs, since unification is equivalent to matching in this case).

For DEDUCE, this depends upon the choice of the theory T modulo which unification is done, but we have the following result which can be applied in each case.

LEMMA 5.5. *When equations are ground, and the theory T used for unification is AC or ACU or ACI or ACUI, then* DEDUCE *keeps the same G_F.*

PROOF. AC-deduction between two ground rules is always of the form (Narendran and Rusinowitch, 1991; Marché, 1991):

$$\left\{ \begin{array}{l} s+t_1 \to u_1 \\ s+t_2 \to u_2 \end{array} \right. \vdash t_1+u_2 = t_2+u_1$$

because superpositions not at top or superpositions at top between rules whose top symbols are not AC are useless, since in such a case COLLAPSE can be applied.

This is also the case for ACU, ACI or ACUI deduction, by the second remark following proof of Theorem 3.8. □

5.2. Termination of completion in several interesting cases

In each case considered in the following, we consider the cases above to insure that the inference rules ORIENT and DEDUCE do not increase $G_F(E,R)$. Then, by assuming that completion does not terminate, we use the previous theorem to build a contradiction. This will use also the following variant of Higman lemma (Higman, 1952).

LEMMA 5.6. *If $\mathcal{E}$ is a finite set, in any infinite sequence $M_1, M_2, M_3, \ldots$ of multi-sets on $\mathcal{E}$ there exists an infinite sequence of indices $i_1,\ i_2, \ldots$ such that $M_{i_1} \subseteq M_{i_2} \subseteq \cdots$.*

We first have to show how to define a total ordering in each case we are interested in. In the case of simple theories ACU, ACI, ACUI, AC0 and ACN, we can use the total AC ordering of Narendran and Rusinowitch (Narendran and Rusinowitch, 1991) or the one of Nieuwenhuis and Rubio (Nieuwenhuis and Rubio, 1993) (with the condition $+ > 0$ for ACN, in order to orient $x + x \to 0$).

In the case of Abelian group theory (AG), 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 ring theory (CR), it is a bit more complicated since the total AC orderings above always orient distributivity in the wrong way! The solution is to use the lexicographic extension of the *modified associative path ordering* (Delor and

Puel, 1993) with precedence $1 > * > - > + > 0$, and then any total AC-compatible ordering (Marché, 1995). Such an ordering can be used also for Boolean ring (BR) and finite fields theories FF(p).

THEOREM 5.7. *If the initial set of equations is ground, then the S-normalized completion terminates when S is either AC, ACU, ACI, ACUI, 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 finite S-normalized canonical rewriting system, in particular it has a decidable word problem.*

PROOF. if S is either AC, ACU, ACI, ACUI, AC0, ACN: if completion does not terminate, we know from theorem 5.4 that there are infinitely many rules whose top symbol is in F, but since there are finitely many symbols, we know in fact that there is a function symbol f in F such that infinitely many rules have f as top symbol in the left-hand side. But F in these cases contains only AC operators and constants, and it is not possible that several rules have a given constant as left-hand side (or else COLLAPSE should be applied), f is necessarily an AC operator, say $+$. Hence we have an infinite sequence of rules $l_1 = l_{1,1} + \cdots + l_{1,k_1} \to r_1, l_2 = l_{2,1} + \cdots + l_{2,k_2} \to r_2, \ldots$ in R_∞. We know that $l_{i,j}$ belongs to the finite set $G_F(R_\infty)$. Applying lemma 5.6 on the multi-sets $M_1 = \{l_{1,1}, \ldots, l_{1,k_1}\}, M_2 = \{l_{2,1}, \ldots, l_{2,k_2}\}, \ldots$ we know that there exist i and j such that $M_i \subseteq M_j$ (modulo AC), hence l_i is a sub-term modulo AC of l_j, hence $l_j \to r_j$ is simplifyable, a contradiction.

if S is AG: the difference between this case and previous ones is that there is also the unary operator $-$ in F, so it is possible that we have in fact infinitely many whose top symbol at left-hand side is $-$. Hence we have an infinite sequence of rules $-l_1 \to r_1, -l_2 \to r_2, \ldots$ in R_∞, but l_i cannot have a $+$ at top because it would not be in S-normal form, hence l_i belongs to the finite set $G_F(R_\infty)$, hence there are i and j such that $l_i = l_j$ and then R_∞ is not reduced.

if S is CR: there is a little difficulty here since the completion generates rules with variables. Lemma 5.5 cannot be applied here since there are rules with variables, but it is still true that G_F does not increase, because of the particular forms of rules we have: deduction between rules whose left-hand sides are either of the form l, $l + \cdots + l$, $x * l + \cdots + x * l$, $-l \to r$ or $-(x * l)$ where l is ground, can produce only ground critical pairs — or pairs of the form $x * s_1 + \cdots + x * s_n = x * t_1 + \cdots x * t_m$ where each s_i and t_j is ground — which have the same G_F. We know then that $G_F(R_\infty)$ is finite. If there are infinitely many rules whose top symbol is $*$ or $-$, we conclude just as before. If there are infinitely many rules whose top symbol is $+$, of the form $l_1 + \cdots + l_k \to r$: thanks to symmetrization we know that the subterms $l_1, \ldots, l_k$ are identical. We know then that we have an infinite sequence of rules $n_1 l_1 \to r_1, n_2 l_2 \to r_2, \ldots$ from which we can extract a sub-sequence such that the top symbols of the l_i's are the same symbol f. If this f is not $*$ we can conclude easily, but if f is $*$ we need Higman Lemma. We know that each l_i is of the form $l_{i,1} * \cdots * l_{i,k_i}$, where each $l_{i,j}$ is in the finite set $G_F(R_\infty)$. By Higman Lemma we can assume without lost of generality (by extracting again a subsequence) that $\{l_{1,1} * \cdots * l_{1,k_1}\} \subseteq \{l_{2,1} * \cdots * l_{1,k_2}\} \subseteq \cdots$, hence l_1 is a subterm of l_2, which is a subterm of l_3, etc. Let us remark finally that the infinite sequence of natural numbers $n_1, n_2, \ldots$ cannot infinitely decrease hence there are i and j such that $n_i \leq n_j$, and then $n_i l_i$ is a subterm of $n_j l_j$, hence COLLAPSE could be applied.

	AC	ACU	AG	AG∪ACU	RRL	REVEAL	REDUX
Computation time	10"18	8"70	1"37	1"42	4"9	22"6	9"50
Number of critical pairs generated	375	299	46	39	108	406	109

Figure 17. Commutative ring theory modulo AC, ACU and AG

if S is BR or FF: we conclude as in the case $S = \mathrm{CR}$ (even more easily since there are no extra variables in these cases). □

Remark: an interesting open question is what property do the S above have in common which insure termination of S-normalized completion? One could think about the fact that each associative operators is also commutative, but this is not sufficient: we will see that it does not work with $S =$ACD.

6. Some implementation results

All the computation times below are of course machine-dependent. We give them for information, but the really significant data are the number of critical pairs computed.

6.1. Commutative rings modulo AC, ACU and AG

We first show what happens when completing commutative ring theory modulo AC, ACU, AG and AG ∪ ACU. Figure 17 shows practical results obtain by C*i*ME and also compares with the other AC completion systems RRL (Kapur and Zhang, 1989), REVEAL (Anantharaman, 1993) and REDUX (Bündgen, 1993). We can see that completion modulo ACU, and moreover AG, are more efficient than AC completion. Our implementation is not as optimized as the other systems above hence AC completion is less efficient, but when completing modulo AG, it is more efficient indeed. The following example shows well why normalized completion is "optimized" 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 ACU-normalized completion, the equation $0+(-0) = 0$ appears to be in Θ, from which you obtain the rule $-0 \to 0$. 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 ACU-completion. Note that this happens even if you still use AC unification, not ACU, as it is the case for the moment in our implementation. In the case of AG-normalized completion, the improvement is even more spectacular as mentioned in Example 4.9.

6.2. A canonical rewriting system for a finitely generated Abelian group

Consider the Abelian group G presented by $E = \{2a - 3b + c = 0, -3a + 2b + 3c = 0, 2a + 2b - 2c = 0\}$ (Lankford et al., 1984). We give the set of equations above to the AG-normalized completion algorithm, and the result is $\{b \to 9a, c \to 25a, 30a \to 0, -a \to 29a\}$. The AG-normalized completion of this system with our implementation takes 5"67 and computes only 11 critical pairs, whereas the AC completion of $E \cup \mathrm{AG}$ by RRL takes 3'27" and computes 837 critical pairs and by REVEAL takes 22" and computes 183 critical pairs. The differences between AC-completions with the different systems are certainly due to different choices in the completion strategy, but of course, the main remark is that AG-normalized completion using symmetrization is the good strategy!

6.3. Computation of a Gröbner basis of a polynomial ideal

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

Example 6.1. *To compute a Gröbner basis of the ideal $(2X^2Y - Y, 3XY^2 - X)$ over $\mathbb{Z}$ (Kandri-Rody and Kapur, 1984) we give to CR-normalized 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:*

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

which corresponds to the Gröbner basis $\{2X^2Y - Y, X^2Y^2 - X^2 + Y^2, 3Y^2 - 2X^2, 2X^3 - X\}$. It takes 18" to be completed by CiME and computes 103 critical pairs. AC-completion with RRL takes 35" and compute 661 critical pairs, takes 5'50" with REDUX and computes 630 critical pairs, and takes 12'30" with REVEAL and computes 1933 critical pairs.

For polynomials with coefficients in a finite field, we have seen this can be done by $FF(p)$-normalized completion.

Example 6.2. *The same example as above over $\mathbb{F}_5$ produces*

$$\left\{\begin{array}{l}
XXX \to 3X \\
YXX \to 3YY \\
YY \to 4XX
\end{array}\right.$$

which corresponds to the Gröbner basis $\{X^3 - 3X, X^2Y - 3Y^2, Y^2 - 4X^2\}$. It takes 4"23 to be completed by CiME and computes 16 critical pairs.

The implementation results above shows of course that normalized completion completion is certainly not as efficient as a dedicated Gröbner bases computation techniques, but that's not surprising at all.

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

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

6.4. Theory of rings homomorphism

Let us finish this list of examples by a CR-normalized completion of a non-ground set of equations. Given two commutative rings with operators $(+, 0, -, \times, 1)$ and $(\oplus, O, \ominus, \otimes, I)$ respectively, the theory of an homomorphism h is given by

$$\begin{cases} h(x+y) = h(x) \oplus h(y) \\ h(x \times y) = h(x) \otimes h(y) \end{cases}$$

The normalized completion modulo $\mathrm{CR}(+, 0, -, \times, 1) \cup \mathrm{CR}(\oplus, O, \ominus, \otimes, I)$ returns the set of rules

$$\begin{cases} h(x+y) \to h(x) \oplus h(y) & h(x \times y) \to h(x) \otimes h(y) \\ h(0) \to O & h(-x) \to \ominus h(x) \\ h(x) \otimes h(1) \to h(x) & \end{cases}$$

Notice that the last rule corresponds to the fact that $h(1)$ is an identity for $\otimes$ on the codomain of h only (the equation $h(1) = I$ is *not* an equational consequence of our specification).

Normalized completion takes 5"5 and computes 54 critical pairs, whereas AC-completion by RRL takes 14" and computes 491 critical pairs. In fact, to be honest we should compare the results obtained by AC-completion and the addition of the results obtained by completion of ACU modulo AC (1 critical pair), AG modulo ACU (135 critical pairs), CR modulo AG (39 critical pairs) and homomorphism modulo CR: the total is 229 critical pairs, still significantly lower than the 491 critical pairs obtained by AC-completion.

We see in fact that normalized completion behaves as a kind of modular completion, and it is significantly more efficient than AC-completion.

7. Conclusion

Figure 18 shows, for various E, known results on decidability or undecidability of the word problem of the 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-normalized completion, so the result is much stronger: every E-ground theory has an E-normalized rewrite system. The undecidability of the word problem for ground theories modulo associativity was proved independently by Post and Markov in 1947 (Markov, 1947; Post, 1947), for group theory it is a result of Novikov in 1955 (Novikov, 1955; Stillwell, 1982) and for ground theories modulo ACD it is a recent result (Marché, 1992).

As a conclusion, we have obtained theoretical results: the unification and the generalization of decidability results, and a new completion algorithm, which generalizes 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 optimized normalizing pairs (Θ, Ψ) of equations and rules. It has also the interesting property that it unifies Knuth-Bendix completion (and its extensions AC completion, ACU-constrained completion) and Buchberger algorithm for computing Gröbner bases.

Future work will be to find other interesting particular theories, like non-commutative

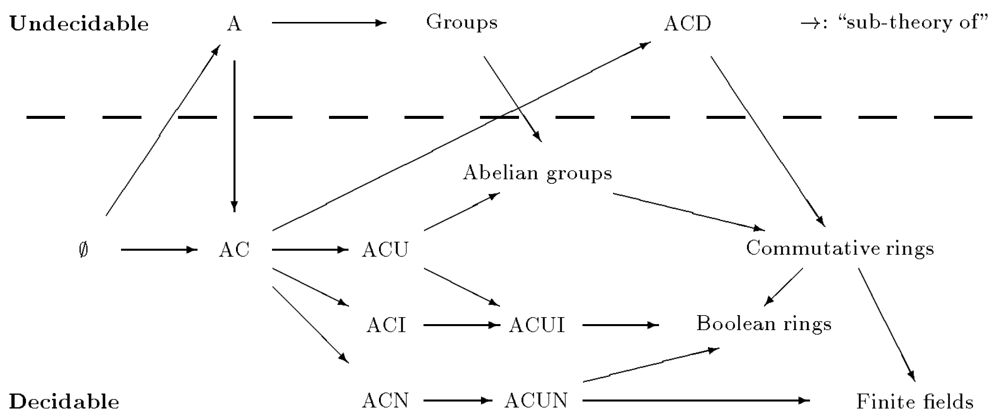

Figure 18. Decidability of the word problem of ground theories modulo E, for some E

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 ACU 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 normalized completion.

ACKNOWLEDGMENTS

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

References

Anantharaman, S. (1993). REVEAL: a users' guide. Rapport de Recherche, Laboratoire d'Informatique Fondamentale d'Orléans.

Bachmair, L. (1991). *Canonical Equational Proofs*. Birkhäuser, Boston.

Bachmair, L. and Dershowitz, N. (1989). Completion for rewriting modulo a congruence. *Theoretical Computer Science*, 67(2&3):173–201.

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

Bachmair, L. and Ganzinger, H. (1994). Buchberger's algorithm: A constraint-based completion procedure. In Jouannaud, J.-P., editor, *First International Conference on Constraints in Computational Logics*, volume 845 of *Lecture Notes in Computer Science*, pages 285–301, München, Germany. Springer-Verlag.

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

Baird, T., Peterson, G., and Wilkerson, R. (1989). 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.

Ben Cherifa, A. and Lescanne, P. (1986). 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.

Birkhoff, G. (1935). On the structure of abstract algebras. In *Proc. Cambridge Phil. Society, 31*.

Book, R. V., editor (1991). *4th International Conference on Rewriting Techniques and Applications*, volume 488 of *Lecture Notes in Computer Science*, Como, Italy. Springer-Verlag.

Buchberger, B. (1965). *An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Ideal*. PhD thesis, University of Innsbruck, Austria. (in German).

Buchberger, B. and Loos, R. (1982). Algebraic simplification. In *Computer Algebra, Symbolic and Algebraic Computation. Computing Supplementum 4*. Springer-Verlag.

Bündgen, R. (1991a). Simulating Buchberger's algorithm by a Knuth-Bendix completion procedure. In (Book, 1991).

Bündgen, R. (1991b). *Term Completion versus Algebraic Completion*. PhD thesis, Universität Tübingen.

Bündgen, R. (1993). Reduce the redex → ReDuX. In Kirchner, C., editor, *5th International Conference on Rewriting Techniques and Applications*, volume 690 of *Lecture Notes in Computer Science*, pages 446–450, Montreal, Canada. Springer-Verlag.

Bürckert, H.-J., Herold, A., Kapur, D., Siekmann, J. H., Stickel, M. E., Tepp, M., and Zhang, H. (1988). Opening the AC-unification race. *Journal of Automated Reasoning*, 4(4):465–474.

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

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

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

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

Domenjoud, E. (1992). Number of minimal unifiers of the equation $\alpha x_1 + \cdots + \alpha x_p \doteq_{AC} \beta y_1 + \cdots + \beta y_q$. *Journal of Automated Reasoning*.

Higman, G. (1952). Ordering by divisibility in abstract algebras. *Proceedings of the London Mathematical Society*, 2(3):326–336.

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

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

Jouannaud, J.-P. and Marché, C. (1992). Termination and completion modulo associativity, commutativity and identity. *Theoretical Computer Science*, 104:29–51.

Kandri-Rody, A. and Kapur, D. (1984). 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.

Kandri-Rody, A., Kapur, D., and Winkler, F. (1989). Knuth-Bendix procedure and Buchberger algorithm — a synthesis. In *Proc. of the 20th Int. Symp. on Symbolic and Algebraic Computation, Portland, Oregon*, pages 55–67.

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

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

Kirchner, C., editor (1990). *Unification*. Academic Press.

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

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

Lankford, D., Butler, G., and Ballantyne, A. (1984). 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.

Lankford, D. S. and Ballantyne, A. M. (1977). 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.

Le Chenadec, P. (1986). *Canonical forms in finitely presented algebras*. Pitman, London.

Loos, R. (1981). Term reduction systems and algebraic algorithms. In *Proceedings of the Fifth GI Workshop on Artificial Intelligence*, pages 214–234, Bad Honnef, West Germany. Available as *Informatik Fachberichte*, Vol. 47.

Marché, C. (1991). On ground AC-completion. In (Book, 1991).

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

Marché, C. (1993). *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 Paris-Sud, Orsay, France.

Marché, C. (1995). Associative-commutative reduction orderings via head-preserving interpretations. Technical Report 95–2, LIFAC, E.N.S. de Cachan.

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

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

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

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

Peterson, G. E. and Stickel, M. E. (1981). Complete sets of reductions for some equational theories. *Journal of the ACM*, 28(2):233–264.

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

Pottier, L. (1989). Algorithmes de complétion et généralisation en logique du premier ordre. Thèse de doctorat. Université de Nice.

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