Centre de Recherche en Informatique de Nancy

Campus Scientifique
Bâtiment LORIA
Boîte Postale 239
54506 Vandœuvre-lès-Nancy Cedex

Téléphone : (+ 33) 3.83.59.20.00
Télécopie : (+ 33) 3.83.41.30.79

Superposition in AC Theories : Proof of Completeness by Semantic Trees

Laurent Vigneron

CRIN 94-R-045

March 1994

Superposition in AC Theories : Proof of Completeness by Semantic Trees

Laurent Vigneron
CRIN-CNRS & INRIA-Lorraine
BP 239, 54506 Vandœuvre-lès-Nancy Cedex, France
E-mail : Laurent.Vigneron@loria.fr

Abstract

We propose an inference system for automated deduction with equality and associative and commutative operators, based on the strategy of superposition. This system is compatible with simplification rules and the associativity and commutativity axioms are handled by the AC-unification algorithm and the inference rules. We prove the refutational completeness of this system by the technique of transfinite semantic trees. Our inference rules simulate the use of classical extended clauses ; so, this avoids any strong control on clauses.

Keywords : Automated Deduction, Associative and Commutative Theories, Resolution, Superposition, Semantic Trees.

Contents

1 Introduction **2**

2 Preliminaries **3**

3 Inference Rules and Lifting Lemma **3**
3.1 Inference Rules 4
3.2 Lifting Lemma 5
3.3 Simplification Rules 6

4 Refutational Completeness of AC Superposition **7**
4.1 Proof Technique 7
4.2 The Quasi-rightmost Path is AC-consistent 12
4.2.1 If K has exactly one successor (Q_α) 14
4.2.2 If K has exactly two successors and Q_α is its left successor 15
4.2.3 If K has exactly two successors and Q_α is its right successor 16
4.3 Proof of the Completeness Theorem 16
4.4 Compatibility with Simplification 20

5 Further Works and Conclusion **22**

References **22**

1 Introduction

Automated deduction with equality and associative commutative (AC) operators (i.e. binary operators f satisfying the axioms : $f(f(x,y),z) \simeq f(x,f(y,z))$ and $f(x,y) \simeq f(y,x)$) has been considered as a difficult problem. The reason is that the presence of AC-axioms increases dramatically the number of possible deductions. For instance, there are 1680 ways to write the term $f(t_1, f(t_2, f(t_3, f(t_4, t_5))))$, where f is an AC-operator and $t_1,\ t_2,\ t_3,\ t_4,\ t_5$ are different constants.

The approach we propose for dealing with AC-axioms is to work in the AC-congruence classes, to employ associative commutative identity checking, pattern matching and unification, and to work only at useful positions (for AC-operators), notion introduced by Lai [Lai89] and applied to completion modulo AC by Domenjoud [Dom91]. This idea of building axioms within the unification procedures has been first initiated by Plotkin [Plo72]. In the context of automated deduction, it has been investigated too by Lankford [Lan79], Stickel [Sti84], Anantharaman and al. [AHM89], Petermann [Pet91]. However these works essentially refer to practical experimentations, and do not account for completeness results of the inference systems that they study. In the following, we focus on giving a complete set of inference rules for first order logic with equality and built-in AC-unification. The recent development of efficient AC-unification algorithms [Sti81, BHK$^+$88] strongly argues in favour of the effectiveness of our method.

Our inference system includes resolution and superposition to deal with equality ; it is an extension of the system defined in [RV91, RV93] for the AC-paramodulation strategy. Superposition, as paramodulation, performs substitution directly by replacing one argument of an equality atom by the other one, when the former occurs in some clause ; it has been introduced by [RW69]. Some important refinements have been proposed by introducing a simplification ordering on terms and forbidding the replacement of a term by a bigger one. These aspects are fully developed in [Pet83, HR91]. Here, we also confine the term replacement step of our superposition rules by a simplification ordering, and we forbid replacements into smaller sides of equations. However, for the sake of completeness we also require this ordering to be total on the set of AC-congruence classes and to be AC-compatible.

The refutational completeness of our set of rules is derived by the transfinite semantic tree method of [Rus89, HR91]. Semantic trees represent the set of Herbrand interpretations for formulas in clausal form. When a tree associated to a formula is empty then we can be sure that the empty clause can be derived, and therefore the initial formula is unsatisfiable. This technique of semantic trees was already used to prove completeness of superposition in the empty theory in [Rus89, Hab90, Rus91]. But, they always had to weaken the superposition strategy by permitting some replacements in right-hand sides of equations or in non-maximal literals. We prove that superpositions are necessary only when applied into maximal side of maximal equational literals.
The superposition strategy has been already extended to AC theories in [Wer92, BG93]. Their inference rules and proof techniques are different from ours, because Wertz, Bachmair and Ganzinger use explicitly extended clauses, which are clauses produced by the associativity axiom ; their proofs are based on the model construction technique of [BG90], but Bachmair and Ganzinger used a special approach, consisting in the construction of an equality model with non-equality models. The inference rules proposed in this paper do not use extended clauses ; they are simulated by inference rules and also we do not need any special control on clauses.

Associative commutative theories have been thoroughly studied in the context of term rewriting systems. We will not review here the Knuth-Bendix method [KB70]. Let us just mention that it has been extended to incorporate associativity and commutativity by [LB77, PS81, JK86].

The layout of this paper is as follows : Section 2 gives some basic material which is relevant to this work. We defined our set of inference rules in Section 3 and prove their lifting lemmas ; this section contains also the definitions of simplification rules. The proof of refutational completeness is developed in Section 4.

2 Preliminaries

This section introduces the basic notions relevant to our work, based on the standard notations and definitions for term rewriting and unification given in [DJ90, JK91].

The *Herbrand universe* is the set of ground terms $\mathcal{T}(\mathcal{F})$, and the *Herbrand base* is the set of ground atoms $\mathcal{A}(\mathcal{P},\mathcal{F})$.
We assume that the operators from a given subset $\mathcal{F}_{AC}$ of $\mathcal{F}$ are associative and commutative, which means that for $f \in \mathcal{F}_{AC}$ the following axioms are implicit in the theory to be considered :

$$AC(f) \quad \begin{cases} f(f(x,y),z) &= f(x,f(y,z)) \\ f(x,y) &= f(y,x) \end{cases}$$

The congruence on $\mathcal{A}(\mathcal{P},\mathcal{F},\mathcal{X}) \cup \mathcal{T}(\mathcal{F},\mathcal{X})$, generated by the associative commutative equations satisfied by the symbols in $\mathcal{F}_{AC}$, is written $=_{AC}$ and is called AC-equality.
A substitution σ is an *AC-unifier* of two objects s and t if $s\sigma =_{AC} t\sigma$. The set of *most general AC-unifiers* (*AC-mgus*) of two terms s and t is defined by : for every AC-unifier θ of s and t, there exists an AC-mgu ρ and a substitution σ such that θ is AC-equal to the composition of σ and ρ (written $\rho\sigma$) ; it means that : for each term u, $u\theta =_{AC} (u\rho)\sigma$.

$\mathcal{MP}os(t)$ denotes the set of all non variable positions p in t such that either $\mathcal{H}ead(t_{|_p}) \notin \mathcal{F}_{AC}$ or, if $p = p'.i$, $\mathcal{H}ead(t_{|_p})$ and $\mathcal{H}ead(t_{|_{p'}})$ are not the same AC-operator. This corresponds to the notion of useful positions defined by Lai [Lai89] and Domenjoud [Dom91].

We denote by $\cap_{AC}$ (resp. $\subseteq_{AC}$) the relation of intersection (resp. inclusion) using the AC-equality for comparing objects. $\subseteq\!\!\!\subseteq_{AC}$ is the multiset extension of $\subseteq_{AC}$.
Given a term t and an AC-operator f, $\mathcal{H}terms(t,f)$ is the multiset of the subterms below f at the root of t ; For instance, $\mathcal{H}terms(a*((a*x)*g(b)),*)$ is $\{a,a,x,g(b)\}$.

Given four terms l_1, l_2, w_1 and w_2 such that $w_1 =_{AC} w_2$, $w_1 = w_1[l_1]$ and $w_2 = w_2[l_2]$, we say that l_1 and l_2 are at *AC-dependent positions* if there is no term w, AC-equal to w_1, such that : $w = w[l_1]_{p_1} = w[l_2]_{p_2}$, and p_1 and p_2 are independent positions, i.e. $p_1 \neq q.p_2$ and $p_2 \neq q.p_1$, for a position q. Such a term w is written $w[l_1]_{p_1}[l_2]_{p_2}$.

Our ordered superposition strategy is based on a *complete simplification ordering* $\succ$ on terms and atoms, total on ground AC-congruence classes (see definition in [RV93]) and AC-compatible [NR91, RN93]. By overloading of notation, given a literal L and a clause D, $L \succ D$ (resp. $L \prec D$) means that the atom corresponding to L is greater than each (resp. less than one) atom of D. The notation $s \not\preceq t$ means that either $s \succ t$ or s and t are incomparable ; on ground objects, by totality, $s \not\preceq t$ is equivalent to $s \succ t$.

The use of the superposition strategy needs to define a new ordering on literals, to express the maximality of a literal in a clause :

Definition 1 *A literal L_1 is greater than a clause D, written $L_1 \succ_L D$, if :*

- *L_1 is not equational and the atom of L_1 is greater than each atom of D,*
- *$L_1 = (l_1 \simeq r_1)$ and for each positive literal $(l_2 \simeq r_2)$ of D, $(l_1 \simeq r_1) \succ (l_2 \simeq r_2)$, and for each negative literal $\neg(l_2 \simeq r_2)$ of D, $(l_1 \simeq r_1) \succ (l_2 \simeq l_2)$ and $(l_1 \simeq r_1) \succ (r_2 \simeq r_2)$,*
- *$L_1 = \neg(l_1 \simeq r_1)$ and for each positive literal $(l_2 \simeq r_2)$ of D, $(l_1 \simeq r_1) \succ (l_2 \simeq \cdot)$ and $(l_1 \simeq r_1) \succ (r_2 \simeq \cdot)$ (where $\cdot$ denotes the empty term), and for each negative literal $\neg(l_2 \simeq r_2)$ of D, $(l_1 \simeq r_1) \succ (l_2 \simeq r_2)$.*

3 Inference Rules and Lifting Lemma

The inference rules, that we shall define in this section, are compatible with the strategy of ordered clauses presented in [Pet83, Rus89]. This strategy permits application of the superposition inference rule under

a refined form : a term cannot be replaced by a more complex one ; in particular, we never superpose into a variable. After the definitions of inferences rules, we shall prove the lifting lemma and present the simplification rules.

The main difference between strategies of paramodulation and superposition is that we never apply replacements into smaller sides of equations.

3.1 Inference Rules

Let us define our inference rules. Note that they require AC-unification.

Definition 2 (AC-Factoring)

$$\frac{L_1 \vee \ldots \vee L_n \vee D}{(L_1 \vee D)\sigma} \quad \textit{if} \quad \left\{ \begin{array}{l} \sigma \in AC_mgus\{L_1, \ldots, L_n\} \\ L_1\sigma \not\preceq_L D\sigma \end{array} \right.$$

Definition 3 (AC-Resolution)

$$\frac{L_1 \vee D_1 \qquad \neg L_2 \vee D_2}{(D_1 \vee D_2)\sigma} \quad \textit{if} \quad \left\{ \begin{array}{l} \sigma \in AC_mgus\{L_1, L_2\} \\ L_1\sigma \not\preceq_L D_1\sigma \vee D_2\sigma \end{array} \right.$$

The next rule simulates a resolution with the reflexivity axiom ($x \simeq x$).

Definition 4 (AC-Reflexion)

$$\frac{\neg(s \simeq t) \vee D}{D\sigma} \quad \textit{if} \quad \left\{ \begin{array}{l} \sigma \in AC_mgus\{s, t\} \\ \neg(s \simeq t)\sigma \not\preceq_L D\sigma \end{array} \right.$$

Definition 5 (AC-Superposition)

$$\frac{(l_1 \simeq r_1) \vee D_1 \qquad L \vee D_2}{(L[r_1]_p \vee D_1 \vee D_2)\sigma} \quad \textit{if} \quad \left\{ \begin{array}{l} \sigma \in AC_mgus\{L_{|_p}, l_1\} \quad \textit{where } p \in \mathcal{MP}os(l_2) \\ (l_1 \simeq r_1)\sigma \not\preceq_L D_1\sigma \\ L\sigma \not\preceq_L (l_1 \simeq r_1)\sigma \vee D_2\sigma \\ l_1\sigma \not\preceq r_1\sigma \\ l_2\sigma \not\preceq r_2\sigma \quad \textit{if } L \textit{ is } (l_2 \simeq r_2) \textit{ or } \neg(l_2 \simeq r_2) \textit{ and } p \textit{ is in } l_2 \end{array} \right.$$

Definition 6 (Contextual AC-Superposition)

$$\frac{(l_1 \simeq r_1) \vee D_1 \qquad L \vee D_2}{(L[f(r_1, z)]_p \vee D_1 \vee D_2)\sigma}$$

$$\textit{if} \quad \left\{ \begin{array}{l} \sigma \in AC_mgus\{L_{|_p}, f(l_1, z)\} \quad \textit{where} \quad f = \mathcal{H}ead(l_1\sigma) \in \mathcal{F}_{AC}, \\ \qquad\qquad\qquad\qquad\qquad\qquad z \textit{ is a new variable, } p \in \mathcal{MP}os(L) \\ (l_1 \simeq r_1)\sigma \not\preceq_L D_1\sigma \quad \textit{and} \quad L\sigma \not\preceq_L D_2\sigma \\ l_1\sigma \not\preceq r_1\sigma \\ l_2\sigma \not\preceq r_2\sigma \quad \textit{if } L \textit{ is } (l_2 \simeq r_2) \textit{ or } \neg(l_2 \simeq r_2) \textit{ and } p \textit{ is in } l_2 \\ \forall w \in \mathcal{H}terms(L_{|_p}, f),\ \mathcal{H}terms(l_1\sigma, f) \not\subseteq_{AC} \mathcal{H}terms(w\sigma, f) \end{array} \right.$$

The condition on $\mathcal{H}terms$ permits to check that the left-hand side l_1 was not found in a variable of $L_{|_p}$.

Definition 7 (Extended AC-Superposition)

$$\frac{(l_1 \simeq r_1) \vee D_1 \qquad (l_2 \simeq r_2) \vee D_2}{((f(r_1, z_1) \simeq f(r_2, z_2)) \vee D_1 \vee D_2)\sigma}$$

$$\textit{if} \quad \begin{cases} \sigma \in AC_mgus\{f(l_1, z_1), f(l_2, z_2)\} \quad \textit{where} \quad f = \mathcal{H}ead(l_1\sigma) = \mathcal{H}ead(l_2\sigma) \in \mathcal{F}_{AC}, \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad z_1 \textit{ and } z_2 \textit{ are new variables} \\ (l_1 \simeq r_1)\sigma \not\preceq_L D_1\sigma \quad \textit{and} \quad (l_2 \simeq r_2)\sigma \not\preceq_L D_2\sigma \\ l_1\sigma \not\preceq r_1\sigma \quad \textit{and} \quad l_2\sigma \not\preceq r_2\sigma \\ \mathcal{H}terms(l_1\sigma, f) \cap_{AC} \mathcal{H}terms(l_2\sigma, f) \neq \emptyset \\ \mathcal{H}terms(z_1\sigma, f) \cap_{AC} \mathcal{H}terms(z_2\sigma, f) \simeq \emptyset \end{cases}$$

The condition on $\mathcal{H}terms$ permits to check that there is a maximal overlap between both left-hand sides l_1 and l_2.

The next rule is needed for completeness. It ¡¡ transforms ¿¿ clauses having two positive equations with the same left-hand side.

Definition 8 (Equational AC-Factoring)

$$\frac{(l_1 \simeq r_1) \vee (l_2 \simeq r_2) \vee D}{((l_1 \simeq r_1) \vee \neg(r_1 \simeq r_2) \vee D)\sigma} \quad \textit{if} \quad \begin{cases} \sigma \in AC_mgus\{l_1, l_2\} \\ (l_1 \simeq r_1)\sigma \not\preceq_L (l_2 \simeq r_2)\sigma \vee D\sigma \\ l_1\sigma \not\preceq r_1\sigma \quad \textit{and} \quad r_1\sigma \not\preceq r_2\sigma \end{cases}$$

Now that inference rules are defined, we have to prove the lifting lemma, since the proof of completeness will be done in the ground case.

3.2 Lifting Lemma

The purpose of the lifting lemma is to lift inferences from the ground level to the general level, i.e. to show that inferences made with ground clauses can be done with the corresponding general clauses. However, lifting superposition rules is often difficult. Let us illustrate this problem by an example : let $P(x, x, c)$ be a clause C and $(c \simeq a)$ an equation, with $c >_p a$; considering the instance $P(c, c, c)$ of $P(x, x, c)$, the superposition from $(c \simeq a)$ into the third argument of $P(c, c, c)$ produces $P(c, c, a)$; an analogous superposition into C generates $P(x, x, a)$, which admits $P(c, c, a)$ as an instance. However, if we superpose in the first argument of $P(c, c, c)$, we obtain $P(a, c, c)$. Since the superposition rule is never applied into a variable, there is no inference between C and $(c \simeq a)$ that can produce a clause which has $P(a, c, c)$ as an instance.

However, if we consider only substitutions which replace variables of $P(x, x, c)$ by irreducible terms, an instance of $P(x, x, c)$ in which we could superpose will be $P(a, a, c)$. Now, every superposition from $(c \simeq a)$ into $P(a, a, c)$ is an instance of a superposition from $(c \simeq a)$ into C. Hence the following definition :

Definition 9 *Let I be an interpretation and σ and θ two ground substitutions. We say that σ is I-reducible to θ, and we write $\sigma \rightarrow_I \theta$, if σ is identical to θ except for a variable x, and $\sigma(x) \longrightarrow_I \theta(x)$. If σ cannot be I-reduced to any substitution, we say that σ is I-irreducible.*

Theorem 10 (Irreducible Substitution Theorem) *Let I be an interpretation, and let $C\sigma$ be a clause whose atoms are all in the domain of I. Then, there exists a ground I-irreducible substitution θ such that $I(C\sigma) = I(C\theta)$.*

The proof of this theorem is detailed in [HR91], and it is based on the noetherianity of the relation $\rightarrow_I$.

This theorem indicates that we only have to consider I-irreducible substitutions. It allows to lift superposition from ground clauses to clauses in general, by considering, in ground clauses, only those positions that already exist in the corresponding general clause.

First, we will use the following property, derived from the stability of ordering $\succ$.

Proposition 11 *Let A and B be two objects (terms or atoms). If $A\sigma \succ B\sigma$ for a substitution σ, then $A \not\preceq B$.*

A consequence of this proposition is :

Corollary 12 *Let σ be a ground AC-unifier of objects $A_1, \ldots, A_m$, and B_1 and B_2 two objects such that : $B_1\sigma \succ B_2\sigma$. There exists an AC-mgu τ of $A_1, \ldots, A_m$ such that : $B_1\tau \not\preceq B_2\tau$.*

This corollary permits to lift ordering conditions of inference rules. The substitution τ is defined by : $\sigma \simeq \rho \circ \tau$, for a ground substitution ρ. The other conditions on $\mathcal{H}terms$ are lifted such as in [RV93].

Lemma 13 (Lifting Lemma) *Let $C_1, \ldots, C_n$ be clauses and σ a ground substitution. If an inference rule R between $C_1\sigma, \ldots, C_n\sigma$ produces a clause C, the same inference rule R can be applied between clauses $C_1, \ldots, C_n$. It generates a clause D, which admits C as a ground instance.*

3.3 Simplification Rules

In this section, we introduce new inference rules that allow to delete redundant clauses, and therefore to reduce the search space of the theorem-proving procedures.

Definition 14 (AC-subsumption) *Let C_1 and C_2 be two clauses satisfying : C_2 has at least as many literals as C_1, and there is a substitution σ such that $C_1\sigma \subseteq_{AC} C_2$, considering a clause as a multiset of literals. Then, we say that the clause C_1 AC-subsumes the clause C_2.*

Definition 15 (Strict AC-subsumption) *Let C_1 and C_2 be two clauses. We say that C_1 strictly AC-subsumes the clause C_2 if C_1 AC-subsumes C_2 and C_2 does not AC-subsumes C_1.*
Application of this rule consists in the deletion of strictly subsumed clauses (C_2).

The result of Loveland [Lov78] about the impossibility of an infinite sequence $C_0, C_1, \ldots$, where each C_{i+1} strictly subsumes C_i, for all i, remains true in AC theories :

Lemma 16 *There is no infinite sequence $C_0, C_1, \ldots$ such that the clause C_{i+1} strictly AC-subsumes the clause C_i, for all i.*

Now, we define two simplification rules, where AC-matching is used. Application of these rules consists in replacing the simplifiable clause by the result of the simplification step.

Definition 17 (AC-simplification)

$$\frac{(l \simeq r) \vee D_1 \qquad L \vee D_2}{L[r\sigma]_p \vee D_2} \quad \textit{if} \quad \begin{cases} L_{|_p} =_{AC} l\sigma \quad \textit{where} \quad p \in \mathcal{MP}os(L) \\ D_2 \simeq D_2' \vee D_2'' \quad \textit{and} \quad D_1\sigma \subseteq_{AC} D_2'' \\ l\sigma \succ r\sigma \\ (l \simeq r)\sigma \prec L \vee D_2' \end{cases}$$

Definition 18 (AC-contextual simplification)

$$\frac{(l \simeq r) \vee D_1 \qquad L \vee D_2}{L[f(r,x)\sigma]_p \vee D_2} \quad \textit{if} \quad \begin{cases} L_{|_p} =_{AC} f(l,x)\sigma \quad \textit{where } f \simeq \mathcal{H}ead(l\sigma) \in \mathcal{F}_{AC}, \\ \qquad\qquad\qquad\qquad x \textit{ is a new variable, } p \in \mathcal{MP}os(L) \\ D_2 \simeq D_2' \vee D_2'' \quad \textit{and} \quad D_1\sigma \subseteq_{AC} D_2'' \\ l\sigma \succ r\sigma \end{cases}$$

A number of other simplification rules are compatible with AC-superposition. For instance, we can delete *tautologies* ; there are two kinds of tautologies : clauses that contain a literal AC-equal to an identity ($s \simeq s$) ; and clauses that contain complementary literals, i.e. literals L and $\neg L'$, where L and L'

are AC-equal.
Another important rule is the *clausal simplification* rule, which deletes all instances of a literal L in every clause of S, if $\neg L$ is a clause of S. We can also delete every instance of $\neg(x \simeq x)$ (it is a kind of trivial reflexion). A clausal simplification step may be viewed as a resolution step, followed by the deletion of one of the parent clauses.

Other reductions are possible, as *reductions by replacement*, i.e. to replace a clause $\neg(s \simeq t) \vee D$, where $s \succ t$, by the clause $\neg(s \simeq t) \vee D'$, where D' is equal to D, except that all subterms AC-equal to s have been replaced by t.

4 Refutational Completeness of AC Superposition

In this section, we shall prove the refutational completeness of the inference rules introduced in the previous section. The main differences between this proof and the proof in the empty theory are the construction of the rightmost maximal path in the semantic tree, and the numerous additional subcases introduced by the associative commutative axioms when considering failure nodes. We need to show that these additional failure nodes can be handled by our set of inference rules.

Let *INF* be the set of inference rules and simplification rules defined earlier. A *derivation* from a set of clauses S_0 is a sequence $S_0, S_1, \ldots$ of sets of constrained clauses obtained by successive applications of rules of *INF*. Let S^* denote $\bigcup_{i \geq 0} S_i$. A derivation $S_0, S_1, \ldots$ is *fair* if each clause C of $\bigcap_{i \geq j} INF(S_i)$ is subsumed by a clause C' of S^*. A clause C is *persistent* in a fair derivation $S_0, S_1, \ldots$ if there is an index k such that C belongs to $\bigcap_{i \geq k} S_i$. A set of clauses S is *AC-unsatisfiable* if it has no *AC-consistent* model, i.e. no model coherent with the AC axioms.

Theorem 19 (Completeness Theorem) *If S is an AC-unsatisfiable set of clauses, then every fair derivation from S generates the empty clause.*

The remaining of this section is devoted to the proof of this Completeness Theorem. In these proofs, we only consider inferences on ground instances of clauses of S^*. For the general case, lifting lemmas described in Section 3.2 can be applied, even for AC-superposition and contextual AC-superposition rules, since it is always possible to label a failure node K by a clause whose variables are instantiated by K-irreducible terms ; hence, superposition can be restricted to non-variable positions. This argument is standard and will not be discussed further.

In the following, we do not present the basic notions on transfinite semantic trees and interpretations (they can be found in [HR91]), but we introduce some new notions used in our proof. After having proved the completeness result, we show that our proof is compatible with simplification.

4.1 Proof Technique

The Herbrand base $\mathcal{A}(\mathcal{P}, \mathcal{F})$ (the set of ground atoms) is ordered as an increasing sequence $\{A_i\}_{i<\lambda}$ by a complete simplification ordering which is AC-compatible, i.e. given two AC-equal atoms A_α and A_γ, each atom A_β between A_α and A_γ has to be AC-equal to A_α. Given an atom A_α, we write W_α the initial segment $\{A_i \mid i < \alpha\}$. Given the ordinal α, an *interpretation* on W_α is a mapping I (sometimes written I_α) from W_α to $\{T, F\}$ satisfying :

- if $(s \simeq s) \in W_\alpha$, then $I(s \simeq s) = T$
- if $(s \simeq t)$, $B[s]$, $B[t] \in W_\alpha$ and $I(s \simeq t) = T$, then $I(B[s]) = I(B[t])$

By the next theorem, it is possible to build inductively all the interpretations. An atom is reducible if it can be rewritten by a smaller equational atom satisfied by the interpretation.

Theorem 20 *[HR91] Let $I : W_{\alpha+1} \to \{T, F\}$ be such that I is an interpretation on W_α. Let J be the restriction of I to W_α. Then, I is an interpretation on $W_{\alpha+1}$ iff :*

(1) A_α *is reducible to an atom* B *and* $I(A_\alpha) \simeq I(B)$ *, or*
(2) A_α *is irreducible, of the form* $(t \simeq t)$ *, and* $I(A_\alpha) \simeq T$ *, or*
(3) A_α *is irreducible and not of the form* $(t \simeq t)$ *.*

Let I and J be two interpretations defined as in the previous theorem. We say that I *is an extension of* J. The collection of all interpretations is called a *transfinite semantic tree*. A *node* is an element of the tree, and the right node of an irreducible atom A_α is the interpretation R such that $R(A_\alpha) = F$.

We define an interpretation I_α on W_α to be *AC-inconsistent* if there are two AC-equal atoms A and B in W_α which are differently interpreted by I_α, i.e. $I_\alpha(A) = T$ and $I_\alpha(B) = F$.

Definition 21 *Let* T *be a transfinite semantic tree. We call* maximal consistent semantic tree *of a set of clauses* S*, and we write MCT(S), the maximal subtree of* T *such that :*

> *For every node* I *in MCT(S), every clause* C' *in* S*, and every clause* C*, AC-equal to a ground instance of* C' *and whose atoms are in the domain of* I*,* $I(C) = T$ *.*

Then, a *failure node* (*FN* for short) is an interpretation I_α falsifying a clause C, AC-equal to a ground instance of a clause C' of S.

Let us define AC by the set of all the AC axioms. The following lemma states that there is no path in $MCT(S \cup AC)$ which is defined on the entire Herbrand base, provided that S is AC-unsatisfiable.

Lemma 22 *Let* S *be a set of clauses ; then,* S *is AC-unsatisfiable if and only if every maximal path in* $MCT(S \cup AC)$ *extends to a failure node.*

Adding AC to S is necessary in order to introduce failure nodes when an interpretation falsifies an equation $(u \simeq u')$ where $u =_{AC} u'$. Such an interpretation may be consistent but not AC-consistent and it should be discarded from the set of potential models of S.

To prove the Completeness Theorem, we reason by contradiction. We assume that S is an AC-unsatisfiable set of clauses and that S^* does not contain the empty clause. Therefore $MCT(S^* \cup AC)$ is not empty. Let us define the sets

$$\begin{aligned} \mathcal{GS} &= \{\ C \mid \exists C' \in S^*\ ,\ C =_{AC} C'\sigma\ ,\ \text{for some ground instance } C'\sigma \text{ of } C'\ \}\ , \\ \mathcal{AC} &= \{\ (u \simeq u') \mid u, u' \in \mathcal{T}(\mathcal{F}),\ u =_{AC} u',\ \mathcal{H}ead(u) \in \mathcal{F}_{AC}\ \}\ \text{and} \\ \mathcal{S}^* &= \mathcal{GS}\ \cup\ \mathcal{AC} \end{aligned}$$

$\mathcal{GS}$ is also the set of all clauses which are AC-equal to a ground instance of a clause of S^*, and $\mathcal{AC}$ is the set of all equalities which are AC-equal to a ground instance of an AC-axiom. By definition of the maximal consistent semantic tree, $MCT(\mathcal{S}^*)$ is equivalent to $MCT(S^* \cup AC)$. Hence, $MCT(\mathcal{S}^*)$ is non empty too.

We can always assume that there is a unique maximal literal in a ground clause, since the factoring rule allows us to eliminate multiple occurrences of this literal. We do not elaborate on this point, since it is quite similar to the standard case [HR91].

The method used for the empty theory is to build a sequence of interpretations, by transfinite induction, following the rightmost path of the maximal consistent semantic tree, then to prove that it is empty and derive a contradiction with the non-emptiness hypothesis of the tree. However, in the present case, the rightmost path may be an *AC-inconsistent path* and should not be considered. We shall instead build the rightmost AC-consistent path. First, let us define what we mean by an *AC-consistent node* in $MCT(\mathcal{S}^*)$.

Definition 23 *Let* K *be an interpretation of* $MCT(\mathcal{S}^*)$*, defined on* W_α*.* K *is AC-inconsistent, if :*

$$\exists B_1, B_2 \in W_\alpha\ ,\ B_1 =_{AC} B_2\ \ \textit{and}\ \ K(B_1) \neq K(B_2)$$

Otherwise, K *is said to be AC-consistent.*

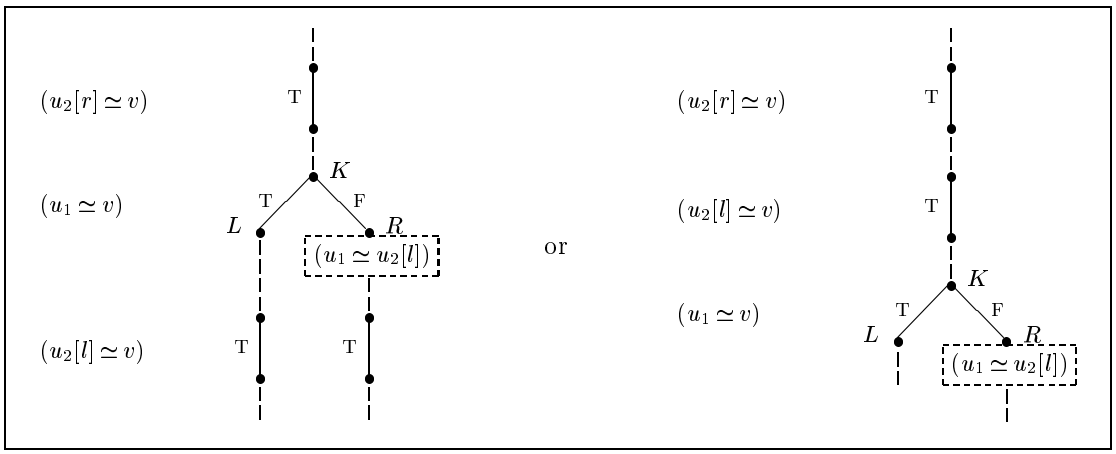


Figure 1: Quasi-Failure Node

We shall need to define an extension of the notion of failure node, quasi-failure node, in order to detect an AC-inconsistency as soon as it occurs on an interpretation. Since an AC-inconsistency may be well-hidden, the notion of quasi-failure node is rather tricky ; it is illustrated by the Figure 1.

Definition 24 (Quasi-Failure Node) *Let K be a node of $MCT(\mathcal{S}^*)$, defined on W_α, such that A_α is an atom $(u_1 \simeq v)$, with $u_1 \succ v$; K admits two extensions L and R : $L(A_\alpha) = T$, $R(A_\alpha) = F$. Then, R is a quasi-failure node (QFN for short) if there is a term $u_2[l]$, AC-equal to u_1, which is reducible by an atom $(l \simeq r)$ to $u_2[r]$, with $l \succ r$ and $K(u_2[r] \simeq v) \simeq T$. The label of a quasi-failure node is defined to be $(u_1 \simeq u_2[l])$.*

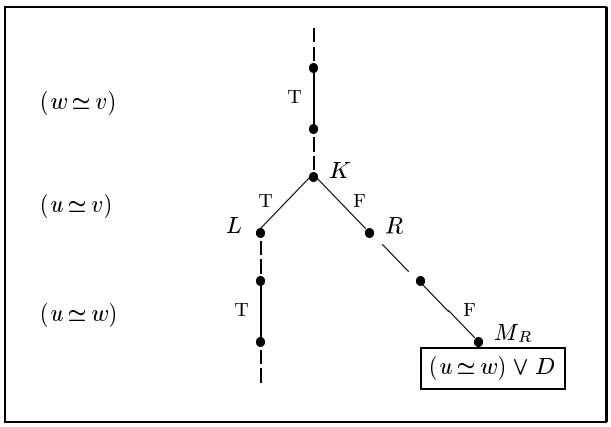


Figure 2: Distant Failure Node

With the superposition strategy, it is forbidden to reduce right-hand sides of equations. So, we have to define a special failure node, called distant failure node and illustrated by the Figure 2, to avoid situations where only the right-hand side of a positive equation is reducible.

Definition 25 (Distant Failure Node) *Let K be a node of $MCT(\mathcal{S}^*)$, defined on W_α, such that A_α is an irreducible equational atom $(u \simeq v)$, with $u \succ v$; K admits two extensions L and R : $L(A_\alpha) = T$, $R(A_\alpha) = F$, and $R \in MCT(\mathcal{S}^*)$. Then, R is a distant failure node (DFN for short) at the level of an atom $(u \simeq w)$ and labeled by a clause $(u \simeq w) \vee D$, if R is not a quasi-failure node, and*

- *either R is the root of a subtree of $MCT(\mathcal{S}^*)$ whose rightmost path ends with a failure node M_R, labeled by a clause $(u \simeq w) \vee D$ and such that : $w \succ v$, $K(w \simeq v) = T$, and $\forall v' < w$, $M_R(u \simeq v') = F$,*

- *or there is a term u', AC-equal to u, such that $(u' \simeq v)$ is irreducible, $K(u' \simeq v) = T$ and the right node of $(u' \simeq v)$ is a distant failure node at the level of an atom $(u' \simeq w)$ and labeled by a clause $(u' \simeq w) \vee D$.*

We define now the quasi-rightmost path of $MCT(\mathcal{S}^*)$ as its rightmost path which does not contain quasi-failure nodes or distant failure nodes.

Definition 26 (Quasi-Rightmost Path) *The quasi-rightmost path (QRP shortly) of $MCT(\mathcal{S}^*)$ is the interpretation Q_γ, defined on W_γ, and built as follows : Q_0 is the empty interpretation ; we assume that Q_i has been defined for all $i < \alpha$; then, we extend the path to Q_α, if possible, by :*

- *If α is a limit ordinal, as in the classical case [HR91], we simply define Q_α by $\bigcup_{i<\alpha} Q_i$. Then, Q_α belongs to $MCT(\mathcal{S}^*)$,*
- *If α is not a limit ordinal, it has a predecessor α^-. Several cases may occur :*
  - *If Q_{α^-} has exactly one successor J, then*
    - *either J is in $MCT(\mathcal{S}^*)$ and Q_α is J,*
    - *or J is not in $MCT(\mathcal{S}^*)$ and γ is α^-,*
  - *If Q_{α^-} has exactly two successors L and R, $L(A_{\alpha^-}) = T$ and $R(A_{\alpha^-}) = F$, then*
    - *If R is a QFN, a FN or a DFN, then*
      - *either L is a FN and γ is α^-,*
      - *or L is not a FN and Q_α is L,*
    - *Else, R is neither a QFN, nor a FN, nor a DFN, and Q_α is R.*

Let R be a distant failure node as described in Definition 25, labeled by a clause C. Assuming that K is AC-consistent, the clause C has the following properties :

Properties of Distant Failure Nodes :

1. C cannot contain a literal $\neg(u \simeq w')$, since by definition each atom $(u \simeq w')$ has to be falsified by M_R.
2. C cannot contain a literal $\neg(u' \simeq w')$ where $u' =_{AC} u$, $w' \neq_{AC} v$ and $(u' = w')$ is irreducible. Indeed, as $(u \simeq w')$ has to be falsified by M_R, if $w' \prec v$, K is AC-inconsistent ; if $w' \succ v$, the right node of $(u' \simeq w')$ is a failure node, and also a similar failure node has to occur at the level of $(u \simeq w')$.
3. If the clause C contains a literal $\neg(u' \simeq w')$ where $u' =_{AC} u$, previous property says that $(w' \simeq v)$ is valid for K and the atom $(u' \simeq w')$ is reducible into $(u' \simeq v)$, an irreducible atom valid for M_R. The right node of this atom $(u' \simeq v)$ cannot be failure or a quasi-failure node because R should be one too. This is therefore a distant failure node, and as this notion is only used for the building of the quasi-rightmost path, $(u' \simeq v)$ is smaller than $(u \simeq v)$ and the clause labeling this node can be used to label M_R, replacing u' by u ; this new clause does not contain the literal $\neg(u' \simeq w')$.
   We conclude that C cannot contain literals $\neg(u' \simeq w')$ where $u' =_{AC} u$.
4. The clause C does not contain any positive literal $(u' \simeq w')$ where $u' =_{AC} u$ and $K(w \simeq w') = T$. Indeed, a sequence of equational AC-factoring steps in the clause $C = (u \simeq w) \vee D$ produces the clause $(u \simeq w) \vee D'$, where each literal $(u' \simeq w')$ was replaced by $\neg(w \simeq w')$. This deduced clause can label M_R.
5. If the clause C contains positive literals $(u' \simeq w')$ where $u' =_{AC} u$ and which do not satisfy previous property, these literals can be replaced in C by literals $(u \simeq w')$, because $(u \simeq w')$ is falsified by M_R by definition.

Property of Failure Nodes :

1. Let Q_α be an AC-consistent interpretation, restriction of Q_γ to W_α, such that A_α is irreducible. If a clause $L \vee D$ labels an extension of Q_α (L is A_α or $\neg A_\alpha$), then L is maximal in $L \vee D$ for the ordering $\succ_L$.
   This property is trivial if A_α is not an equational atom or if $L = \neg A_\alpha$. But, if $L = A_\alpha = (u \simeq w)$ ($u \succ w$), we have to show that D does not contain a literal $\neg(u' \simeq v)$ where $u' =_{AC} u$. This case cannot happen because such an atom $(u' \simeq v)$ would be valid for Q_α, as $(u \simeq v)$ by AC-consistency, and $(u \simeq w)$ should be reducible by $(u \simeq v)$.

The next proposition shows that, if an atom A_α is Q_α-reducible by an equality $(l \simeq r)$, with $l \succ r$, then there is an AC-equivalent atom A_β which is Q_α-reducible by an equality $(g \simeq d)$ whose right node is a failure or a distant failure node labeled by a clause of $\mathcal{GS}$. This proposition will simplify the proofs of the following propositions.

Proposition 27 *Let Q_α be the restriction of the quasi-rightmost path Q_γ to W_α. If there is a Q_α-irreducible atom $(l \simeq r)$ such that $l \succ r$ and $A_\alpha \longrightarrow^{l \simeq r}_{Q_\alpha} A_\alpha[r]$, then there are two atoms A_β and $(g \simeq d)$, such that $A_\beta =_{AC} A_\alpha$, $g \succ d$, $(g \simeq d)$ is Q_α-irreducible, $A_\beta \longrightarrow^{g \simeq d}_{Q_\alpha} A_\beta[d]$, $Q_\alpha(A_\beta[d]) \simeq Q_\alpha(A_\alpha[r])$, the right node of $(g \simeq d)$ is a failure node or a distant failure node labeled by a clause C_g of $\mathcal{GS}$, and C_g does not contain a literal $\neg(g' \simeq d')$ where g' is AC-equal to g.*

Proof: *Suppose that the proposition is true for all atoms less than but not AC-equal to A_α. Since $(l \simeq r)$ is Q_α-irreducible, it admits two extensions. Let R_l be the right node of $(l \simeq r)$: $R_l(l \simeq r) \simeq F$.*

- *If R_l is a failure node, then it cannot falsify a clause of $\mathcal{AC}$, since $l \succ r$ implies that $l \neq_{AC} r$. Thus, in this case, we can take $(l \simeq r)$ for $(g \simeq d)$ and A_α for A_β.*
- *If R_l is a distant failure node, $(g \simeq d)$ is $(l \simeq r)$ and A_β is A_α.*
- *If R_l is a quasi-failure node, it is labeled by an atom $(l \simeq l'[a])$; $(l'[a] \simeq r)$ is reducible by an irreducible equality $(a \simeq b)$ to $(l'[b] \simeq r)$, where $a \succ b$, and $R_l(l'[b] \simeq r) = Q_\alpha(l'[b] \simeq r) = T$. So, the atom $A_\alpha[l'[a]]$, which is AC-equal to $A_\alpha[l]$, is Q_α-reducible by $(a \simeq b)$ to $A_\alpha[l'[b]]$, and $Q_\alpha(A_\alpha[l'[b]]) = Q_\alpha(A_\alpha[r])$ since $A_\alpha[l'[a]]$ is Q_α-reducible by $(l'[a] \simeq r)$ to $A_\alpha[r]$.*
  By hypothesis, since $(l'[a] \simeq r) \prec A_\alpha$, we can assume that there are two atoms $(l''[a'] \simeq r)$ and $(a' \simeq b')$ such that $l''[a'] =_{AC} l'[a]$, $a' \succ b'$, $(a' \simeq b')$ is Q_α-irreducible, $(l''[a'] \simeq r) \longrightarrow^{a' \simeq b'}_{Q_\alpha} (l''[b'] \simeq r)$, $Q_\alpha(l''[b'] \simeq r) = Q_\alpha(l'[b] \simeq r) = T$, and the right node of $(a' \simeq b')$ is a failure or distant failure node labeled by a clause of $\mathcal{GS}$.
  So, we can take $(a' \simeq b')$ for $(g \simeq d)$, and $A_\alpha[l''[a']]$ for A_β.

In addition, we have to prove that there is no literal $\neg(g' \simeq d')$, where g' is AC-equal to g, in the clause labeling the right node of $(g \simeq d)$. This property is essential to guaranty the maximality in the clause C_g of the atom at the level of which there is a failure node under the right node of $(g \simeq d)$; this atom is $(g \simeq d)$ if the right node of $(g \simeq d)$ is a failure node, or $(g \simeq w)$ if it is a distant failure node.
By definition, this property is true if the right node of $(g \simeq d)$ is a distant failure node.
So, let us assume that there is a minimal atom $(g' \simeq d')$ which does not satisfy this property when the right node of $(g \simeq d)$ is a failure node. This atom cannot be reducible by an AC-axiom because it was chosen minimal. Moreover, if it is reducible by another equation, the right node of $(g \simeq d)$ should be a quasi-failure node.
Hence, $(g' \simeq d')$ is irreducible. Since it is smaller than $(g \simeq d)$ and valid for Q_α, its right node is either a failure node or a distant failure node. Then, our final atoms are $(g' \simeq d')$ and $A_{\alpha''}$, where $A_{\alpha''}$ is the atom $A_{\alpha'}$ in which g was replaced by g'. □

We have to prove another property : if a (contextual) superposition is applied between clauses labeling failure nodes along the quasi-rightmost path, a failure node should cut this path.

Proposition 28 *Let $C_1 = (l \simeq r) \vee D_1$ and $C_2 = L_2 \vee D_2$ be two clauses of $\mathcal{GS}$ labeling failure or distant failure nodes along the quasi-rightmost path of $MCT(\mathcal{S}^*)$, and such that there is a literal L, AC-equal to L_2, Q_γ-reducible by $(l \simeq r)$ $(l \succ r)$. Then, if the ordering conditions for applying a (contextual) superposition from C_1 into C_2 are satisfied, Q_γ falsifies a clause of $\mathcal{GS}$.*

Proof: *The literal $(l \simeq r)$ and L_2 being maximal in they clause, Q_γ falsifies D_1 and D_2. Moreover, L being reducible by $(l \simeq r)$ into $L[r]$, the atoms corresponding to literals L and $L[r]$ have the same truth value (the value of the atom of L_2).*
Let $C'_1 = (l' \simeq r') \vee D'_1$ and $C'_2 = L'_2 \vee D'_2$ be the clauses of S^* admitting respectively C_1 and C_2 as ground instances for a substitution σ. L is reducible by $(l \simeq r)$, so there is a position o in $L'_2\sigma$ such that : $o \in \mathcal{MP}os(L'_2\sigma)$ and $L'_2\sigma_{|_o} =_{AC} l'\sigma$ or the head operator of l is AC (written f) and $L'_2\sigma_{|_o} =_{AC} f(l'\sigma, u)$ for a ground term u.

- *If $L'_2\sigma_{|_o}$ is AC-equal to $l'\sigma$, an AC-superposition from $C'_1\sigma$ into $C'_2\sigma$ at position o generates the clause $L'_2\sigma[r'\sigma]_o \vee D'_1\sigma \vee D'_2\sigma$.*
- *If $L'_2\sigma_{|_o}$ is AC-equal to a term $f(l'\sigma, u)$, a contextual AC-superposition from $C'_1\sigma$ into $C'_2\sigma$ at position o generates the clause $L'_2\sigma[f(r'\sigma, u)]_o \vee D'_1\sigma \vee D'_2\sigma$.*

In both cases, the deduced clause is AC-equal to $L[r] \vee D_1 \vee D_2$, so this clause belongs to $\mathcal{GS}$. But, Q_γ falsifies $L[r]$, D_1 and D_2, i.e. the deduced clause. We conclude that Q_γ cannot be in $MCT(\mathcal{S}^*)$. □

We can note that in previous proof, the position o was chosen in $L'_2\sigma$, but it is a position in L'_2 because the Lifting Lemma says that we have to consider only ground substitutions.

4.2 The Quasi-rightmost Path is AC-consistent

The next proposition says that the quasi-rightmost path is non-empty and AC-consistent.

Proposition 29 *If $MCT(\mathcal{S}^*)$ is non empty, the quasi-rightmost path Q_γ of $MCT(\mathcal{S}^*)$ is non empty and AC-consistent.*

Proof: *First, Q_γ is non empty : by construction of the semantic tree, A_0 is an atom $(a \simeq a)$ and $Q_1(A_0) = T$; so, Q_1 cannot be a failure or a quasi-failure node.*

Second, we have to prove that Q_γ is AC-consistent. We reason by contradiction : if Q_γ is AC-inconsistent, there exists a minimal ordinal α such that Q_α is AC-inconsistent. Necessarily, α **is not a limit ordinal.** *Otherwise, as in the classical case [HR91], if α is a limit ordinal, Q_α is defined by $\bigcup_{i<\alpha} Q_i$, and some Q_i is AC-inconsistent ; this is impossible since α is minimal.*

Since α is not a limit ordinal, α has a predecessor α^-. Let K be the interpretation Q_{α^-}, and let B denote A_{α^-}. By minimality of α, K is AC-consistent. Then,

$$\exists A_\beta <_A B \ , \ A_\beta =_{AC} B \ \text{ and } \ Q_\alpha(B) \neq Q_\alpha(A_\beta)$$

We prove in lemma 30 that B is an equality $(u_2 \simeq v)$ and A_β is an equality $(u_1 \simeq v)$, where $u_1 =_{AC} u_2$ and $u_2 \succ v$. From now on, we assume that $A_\beta = (u_1 \simeq v)$ is the smallest atom such that $A_\beta =_{AC} B$ and $Q_\alpha(A_\beta) \neq Q_\alpha(B)$. Then, several cases are possible, and the next lemmas show that they all yield a contradiction :

1. *If K has exactly one successor, Q_α (paragraph 4.2.1) : B is K-reducible, A_β is K-irreducible and $K(A_\beta) = F$. It means that a quasi-failure node should occur at A_β.*
2. *If K has two successors L and R, at least one of them is in the quasi-rightmost path (Q_α), $L(B) = T$ and $R(B) = F$.*
   (a) *If Q_α is L (paragraph 4.2.2) : $K(A_\beta) = F$ and R is a quasi-failure node, i.e. there is an atom AC-equal to B which is reducible by an equation $(g \simeq d)$ whose right node is either a failure or a distant failure node. A_β is reducible, too, by an equation $(l \simeq r)$ whose right node is a (distant) failure node. From this, we deduce that Q_α cannot belong to $MCT(\mathcal{S}^*)$ because it falsifies a clause of $\mathcal{GS}$ produced by an extended AC-superposition between the clauses labeling right nodes of $(g \simeq d)$ and $(l \simeq r)$.*

(b) *If* Q_α *is* R *(paragraph 4.2.3) :* $K(A_\beta) = T$ *and either* A_β *is reducible, which implies that* R *should be a quasi-failure node, or* A_β *is irreducible and its right node is a failure, distant failure or quasi-failure node, and it should be one for* R *too.*

Since all cases are impossible, Q_α *cannot belong to the quasi-rightmost path* Q_γ*, and also this sequence of interpretations is AC-consistent.* □

Lemma 30 *B is an equality* $(u_2 \simeq v)$ *and* A_β *is an equality* $(u_1 \simeq v)$*, where* $u_1 =_{AC} u_2$ *and* $u_2 \succ v$.

Proof: *We decompose the proof of this lemma in three facts :*

1. $\mathcal{H}ead(B)$ **is "**$\simeq$**"** *; otherwise* $B = P(s_1, \ldots, s_n)$, $A_\beta = P(t_1, \ldots, t_n)$, $s_i =_{AC} t_i$ *for each* i*, and there is a* j *such that* $K(s_j \simeq t_j) = F$ *; so,* K *should be AC-inconsistent.*
   Thus, in the following, we can assume : $B = (u_2 \simeq v_2)$ *and* $A_\beta = (u_1 \simeq v_1)$ *, with* $u_2 \geq v_2$, $u_1 \geq v_1$, $u_1 =_{AC} u_2$ *and* $v_1 =_{AC} v_2$.
2. $u_2 \neq_{AC} v_2$ *; indeed, assuming that* $Q_\alpha(B) = F$ *, if* $u_2 =_{AC} v_2$ *, either* $(u_2 \simeq v_2) \in \mathcal{AC}$ *(if* $\mathcal{H}ead(u_2) \in \mathcal{F}_{AC}$ *), or there is a position in* u_2 *such that* $u_{2|_p} =_{AC} v_{2|_p}$, $\mathcal{H}ead(u_{2|_p}) \in \mathcal{F}_{AC}$ *(i.e.* $(u_{2|_p} \simeq v_{2|_p}) \in \mathcal{AC}$ *) and* $K(u_{2|_p} \simeq v_{2|_p}) = F$ *. In both cases, an equation of* $\mathcal{AC}$ *is falsified by* K*, and also* K *should not belong to* $MCT(\mathcal{S}^*)$*.*
   If $Q_\alpha(B) = T$ *, the reasoning is similar on* $(u_1 \simeq v_1)$*, since in that case* $Q_\alpha(A_\beta) = F$.
   So, $u_2 \succ v_2$ *, and by AC-compatibility,* $u_1 \succ v_1$.
3. **We can choose** A_β **such that** $v_1 = v_2$ *, since if* $v_1 \neq v_2$, $K(v_1 \simeq v_2) = T$, $(u_1 \simeq v_1) \longrightarrow_K^{v_1 \simeq v_2} (u_1 \simeq v_2)$ *and* $Q_\alpha(B) \neq K(u_1 \simeq v_2)$ *; then, we can take* $(u_1 \simeq v_2)$ *for* A_β*. Indeed,* K *is AC-consistent means that* B *cannot be* K*-reduced to* $(u_2 \simeq v_1)$ *by* $(v_2 \simeq v_1)$ $(K(u_1 \simeq v_1) = K(u_2 \simeq v_1))$*, and also* $v_1 > v_2$.

So, B *is an equality* $(u_2 \simeq v)$ *and* A_β *is an equality* $(u_1 \simeq v)$ *such that :* $u_1 =_{AC} u_2$ *and* $u_2 \succ v$. □

The following lemma will be used to prove next lemmas. It states that, for a node K of the quasi-rightmost path, if two AC-equal atoms, taking a different value for K, are K-reducible by different equalities at different positions, then a failure node should occur earlier in the path, provided that K is AC-consistent.

Lemma 31 *Let* $(u_1 \simeq v)$ *and* $(u_2 \simeq v)$ *be two AC-equal atoms such that* $u_1 \succ v$ *. Let us assume that* $(u_1 \simeq v)$ *(resp.* $(u_2 \simeq v)$*) is* K*-reducible by* $(g_1 \simeq d_1)$ *(resp.* $(g_2 \simeq d_2)$*) into* $(u_1[d_1] \simeq v)$ *(resp.* $(u_2[d_2] \simeq v)$*), where* $g_1 \succ d_1$ *and* $g_2 \succ d_2$.
If K *is AC-consistent and* $K(u_1[d_1] \simeq v) \neq K(u_2[d_2] \simeq v)$ *,* K *falsifies a clause of* $\mathcal{GS}$*.*

Proof: *By Proposition 27, there exist atoms AC-equal to* $(u_1 \simeq v)$ *and* $(u_2 \simeq v)$*, keeping interpretations, which are reducible by equations whose right nodes are failure or distant failure nodes labeled by clauses of* $\mathcal{GS}$*. So, let* $(w_1 \simeq v)$*,* $(w_2 \simeq v)$*,* $(l_1 \simeq r_1)$ *and* $(l_2 \simeq r_2)$ *be these atoms.*

$$K(u_1[d_1] \simeq v) = K(w_1[r_1] \simeq v) \;\neq\; K(w_2[r_2] \simeq v) = K(u_2[d_2] \simeq v)$$

Since $K(w_1[r_1] \simeq v) \neq K(w_2[r_2] \simeq v)$ *,* $(w_1[r_1] \simeq w_2[r_2])$ *is falsified by* K*. From this, we can deduce the following facts :*

1. l_1 **and** l_2 **are at AC-dependent positions** *in the AC-congruence class of* w_1*, i.e. there is no term* $w[l_1]_{p_1}[l_2]_{p_2}$*, AC-equal to* w_1*, such that :* $w[r_1]_{p_1}[l_2]_{p_2} =_{AC} w_1[r_1]$ *and* $w[l_1]_{p_1}[r_2]_{p_2} =_{AC} w_2[r_2]$ *. Indeed, the atom* $(w[l_1]_{p_1}[r_2]_{p_2} \simeq w[r_1]_{p_1}[l_2]_{p_2})$ *is valid for* K *since it is reducible to the trivial atom* $(w[r_1]_{p_1}[r_2]_{p_2} \simeq w[r_1]_{p_1}[r_2]_{p_2})$ *, but it is AC-equal to* $(w_1[r_1] \simeq w_2[r_2])$ *and these atoms have a different value for* K*. It would mean that* K *is AC-inconsistent.*
2. l_1 **and** l_2 **are not AC-equal**. *If* l_1 *and* l_2 *were AC-equal, we would have* $K(l_1 \simeq r_2) = K(l_2 \simeq r_1) = T$ *; but, since* $(l_1 \simeq r_1)$ *is not reducible by* $(l_1 \simeq r_2)$*,* r_2 *has to be greater than or equal to* r_1 *; on the other hand,* r_1 *has to be greater than or equal to* r_2 *because* $(l_2 \simeq r_2)$ *is not reducible by* $(l_2 \simeq r_1)$*. The conclusion is* $r_1 = r_2$*. The terms* $w_1[r_1]$ *and* $w_2[r_2]$ *are also AC-equal, but the atom* $(w_1[r_1] \simeq w_2[r_2])$ *being falsified by* K*,* K *should be AC-inconsistent.*

3. l_1 **is not a proper subterm of a** l_3**, AC-equal to** l_2 *; indeed,* $(l_3 \simeq r_2)$ *would be* K*-reducible by* $(l_1 \simeq r_1)$*, and also the failure node at the right node of* $(l_2 \simeq r_2)$ *would be a quasi-failure node, labeled by the atom* $(l_2 \simeq l_3)$ *instead of a clause of* $\mathcal{GS}$*.*

4. *In the same way,* l_2 **is not a proper subterm of a** l_3**, AC-equal to** l_1*.*

From these four facts, we can deduce that there are positions q_1 *in* w_1 *and* q_2 *in* w_2 *such that :* $w_{1|_{q_1}} =_{AC} f(l_1, t_1) =_{AC} f(l_2, t_2) =_{AC} w_{2|_{q_2}}$ *, where* f *is the AC-operator at the head of* l_1 *and* l_2*. Since* l_1 *and* l_2 *are at dependent positions, but none of them is a subterm of the other, there are ground terms* a*,* b *and* c *such that :* $l_1 =_{AC} f(a, b)$ *,* $l_2 =_{AC} f(a, c)$ *, and*

$$\begin{aligned} \mathcal{H}terms(l_1, f) \cap_{AC} \mathcal{H}terms(l_2, f) &= \mathcal{H}terms(a, f) \\ \mathcal{H}terms(b, f) \cap_{AC} \mathcal{H}terms(c, f) &= \emptyset \end{aligned}$$

We can notice that t_1 *and* t_2 *are respectively* c *and* b*, or* $f(c, d)$ *and* $f(b, d)$ *for a ground term* d*.*

Let C_1 *and* C_2 *be clauses of* $\mathcal{GS}$ *labeling the right nodes of* $(l_1 \simeq r_1)$ *and* $(l_2 \simeq r_2)$*. Then, denoting* $C'_1 = (l'_1 \simeq r'_1) \vee D'_1$ *and* $C'_2 = (l'_2 \simeq r'_2) \vee D'_2$ *the clauses of* S^* *such that* $C_1 =_{AC} C'_1\sigma$ *and* $C_2 =_{AC} C'_2\sigma$ *for a ground substitution* σ*, all conditions for applying an* **extended AC-superposition** *step between clauses* $C'_1\sigma$ *and* $C'_2\sigma$ *are satisfied.*
The deduced clause C *is* $(f(r'_1\sigma, c) \simeq f(r'_2\sigma, b)) \vee D'_1\sigma \vee D'_2\sigma$ *and belongs to* $\mathcal{GS}$*.*

We have to show that K *falsifies this clause* C*. Atoms* $(l_1 \simeq r_1)$ *and* $(l_2 \simeq r_2)$ *being maximal in their clause,* K *falsifies* $D'_1\sigma$ *and* $D'_2\sigma$*. So we only have to show that* K *falsifies the atom* $(f(r'_1\sigma, c) \simeq f(r'_2\sigma, b))$*. Let us denote* $wrq1$ *the subterm* $w_1[r_1]_{|_{q_1}}$ *and* $wrq2$ *the subterm* $w_2[r_2]_{|_{q_2}}$*. Since* $wrq1 =_{AC} f(r_1, t_1)$ *and* $wrq2 =_{AC} f(r_2, t_2)$ *, and by AC-consistency of* K*,* K *satisfies equations* $(wrq1 \simeq f(r_1, t_1))$ *and* $(wrq2 \simeq f(r_2, t_2))$*.*
In addition, if we assume that K *satisfies* $(f(r'_1\sigma, c) \simeq f(r'_2\sigma, b))$*, we have :*

$$K(f(r_1, c) \simeq f(r_2, b)) \;=\; K(f(r_1, t_1) \simeq f(r_2, t_2)) \;=\; K(wrq1 \simeq wrq2) \;=\; T$$

Then, whether $wrq1$ *is smaller or bigger than* $wrq2$*, the atom* $(w_1[r_1] \simeq w_2[r_2])$*, falsified by* K*, is* K*-reducible to either* $(w_1[r_1] \simeq w_2[wrq1]_{q_2})$ *or* $(w_1[wrq2]_{q_1} \simeq w_2[r_2])$*. In both cases, the left-hand and right-hand sides of the deduced equation are AC-equal, and since these equations are falsified by* K*,* K *would have to be AC-inconsistent.*
The hypothesis that K *satisfies* $(f(r'_1\sigma, c) \simeq f(r'_2\sigma, b))$ *was also wrong, and we conclude that* K *falsifies the clause* C *of* $\mathcal{GS}$*, produced by the extended AC-superposition step described above, and this interpretation is not in* $MCT(\mathcal{S}^*)$*.* □

4.2.1 If K has exactly one successor (Q_α)

If the interpretation K has a unique successor Q_α, we are going to show that Q_α cannot belong to the quasi-rightmost path. First, let us show that the atom $(u_2 \simeq v)$ (B) is K-reducible to $(u_2[r] \simeq v)$ by an equation $(l \simeq r)$ where $l \succ r$ (Lemma 32), et $(u_1 \simeq v)$ (A_β) is K-irreducible (Lemma 33).

Lemma 32 $B = (u_2 \simeq v) \longrightarrow_K^{l \simeq r} (u_2[r] \simeq v)$ *with* $l \succ r$ *.*

Proof: *In the following, we prove that* $B = (u_2 \simeq v)$ *is* K*-reducible by an equality* $(l \simeq r)$*,* l *is not AC-equal to* r*, and* l *is a subterm of* u_2*.*

1. B **is** K**-reducible by some** $(l \simeq r)$ *; by construction of the semantic tree (Theorem 20),* K *has a single extension implies that either* B *is reducible or* B *is an atom* $(u \simeq u)$ *and* $K(B) = T$*. Since* $u_2 \succ v$*,* B *has to be reducible. Let* $(l \simeq r)$ *be a* K*-irreducible equation reducing* B *:* $B \longrightarrow_K^{l \simeq r} B[r]$ *.*

2. $l \neq_{AC} r$ *because, by AC-consistency of* K*,* B *cannot be reduced to an AC-equal atom, which would also be interpreted differently from* A_β *too. So* $l \succ r$*.*

3. B **is K-reducible at a position in** u_2 ; *indeed, if B was reducible at a position in v, A_β would be reducible in v too, and both reduced terms would be AC-equal but with a different value for K. It is impossible since K is assumed to be AC-consistent.*

Thus, $B \longrightarrow_K^{l \simeq r} (u_2[r] \simeq v)$ *with* $l \succ r$. □

Lemma 33 *A_β is K-irreducible.*

Proof: *If A_β is K-reducible by an equation $(g \simeq d)$, g and d are not AC-equal by minimality of A_β ; so $g \succ d$. We know that B is K-reducible by an equation $(l \simeq r)$ with $l \succ r$. Lemma 31 says that in this case K falsifies a clause of $\mathcal{GS}$, which contradicts its membership to the quasi-rightmost path. We also conclude that A_β is K-irreducible.* □

Lemma 34 *K falsifies A_β and Q_α cannot be a node of the quasi-rightmost path.*

Proof: *We are going to show that a quasi-failure node should occur at A_β, and also that Q_α should not belong to the quasi-rightmost path. But first, let us show that A_β is falsified by K.*
If we assume that A_β is valid for K, and also that B is falsified by Q_α, the right node R of A_β is

- *either a failure node labeled by a clause $C_1 = (u_1 \simeq v) \vee D_1$ of $\mathcal{GS}$; therefore, $(u_2 \simeq v) \vee D_1$ belongs to $\mathcal{GS}$ too and is falsified by Q_α ; so Q_α should not be in $MCT(\mathcal{S}^*)$.*
- *or a quasi-failure node labeled by an equation $(u_1 \simeq u_3[g])$ such that $u_1 =_{AC} u_3[g]$ and $(u_3[g] \simeq v)$ is K-reducible by an equation $(g \simeq d)$ into $(u_3[d] \simeq v)$ $(g \succ d)$. By definition of a quasi-failure node, $R(u_3[d] \simeq v)$ is different from $R(u_1 \simeq v)$; it means that $R(u_3[d] \simeq v) = T$. But, lemma 31 can be applied to atoms B and $(u_3[g] \simeq v)$, and it says that K falsifies a clause of $\mathcal{GS}$, and also cannot be a node of the quasi-rightmost path.*
- *or a distant failure node at the level of an atom $(u_1 \simeq w)$ and labeled by a clause $C_1 = (u_1 \simeq w) \vee D_1$ of $\mathcal{GS}$ at the node M_R. Since B is K-reducible, Proposition 27 permits to find an atom AC-equal to B and interpreted as B, which is reducible by an equation whose right node is a failure or distant failure node. Let C_2 be the clause labeling this node. By Proposition 28, M_R falsifies a clause of $\mathcal{GS}$, produced by a superposition from C_2 into C_1.*

Finally, A_β is K-irreducible and $K(A_\beta) = F$. But, as B is K-reducible by $(l \simeq r)$, $l \succ r$ and $Q_\alpha(B) = T$, R should be a quasi-failure node, and Q_α should not belong to the quasi-rightmost path. □

4.2.2 If K has exactly two successors and Q_α is its left successor

If K has two extensions L and R, $L(B) = T$ and $R(B) = F$, and Q_α is the left extension, i.e. L, we are going to show that Q_α cannot belong to the quasi-rightmost path.

Lemma 35 *Q_α cannot be a node of the quasi-rightmost path.*

Proof: *As Q_α is the left extension of K, $Q_\alpha(B) = T$, $K(A_\beta) = F$, and the right extension R is either a failure node, or a distant failure node, or a quasi-failure node. From this, we deduce that*

1. R **is not a failure node**, *because R would be labeled by a clause $B \vee D_R$ of $\mathcal{GS}$ and the clause $A_\beta \vee D_R$ would belong to $\mathcal{GS}$ too. However, K falsifies this last clause and should not be in $MCT(\mathcal{S}^*)$.*
2. R **is not a distant failure node** ; *indeed, if R was a distant failure node, it would be labeled by a clause $(u_2 \simeq w) \vee D_2$ of $\mathcal{GS}$ at the level of an atom $(u_2 \simeq w)$, and $K(w \simeq v) = T$. But, u_1 is smaller than u_2 and $K(u_1 \simeq w) = K(u_1 \simeq v) = F$. The clause $(u_1 \simeq w) \vee D_2$ of $\mathcal{GS}$ should label a failure node at the level of the atom $(u_1 \simeq w)$, i.e. before the failure node at $(u_2 \simeq w)$.*
   So, R is a quasi-failure node labeled by an equation $(u_2 \simeq u_3[g])$: u_2 and $u_3[g]$ are AC-equal, $(u_3[g] \simeq v) \longrightarrow_K^{g \simeq d} (u_3[d] \simeq v)$ and $g \succ d$; moreover, $K(u_3[d] \simeq v) = T$.

3. A_β **is K-reducible**, *because if it was irreducible, a quasi-failure node should occur at its level since it is falsified by K and the atom $(u_3[g] \simeq v)$ is an AC-equal A_β and reducible. So :* $A_\beta \longrightarrow_K^{l \simeq r} (u_1[r] \simeq v)$ *, with* $l > r$.

4. $l \neq_{AC} r$, *by minimality of* A_β.

To summarize, we have :

- $(u_1 \simeq v) \longrightarrow_K^{l \simeq r} (u_1[r] \simeq v)$ *and* $l \succ r$,
- $(u_3 \simeq v) \longrightarrow_K^{g \simeq d} (u_3[d] \simeq v)$ *and* $g \succ d$,
- $K(u_1[r] \simeq v) \neq K(u_3[d] \simeq v)$.

Lemma 31 can be applied, and we deduce that K falsifies a clause of $\mathcal{GS}$ and should not be in $MCT(\mathcal{S}^*)$, which implies that Q_α is not a node of the quasi-rightmost path. □

4.2.3 If K has exactly two successors and Q_α is its right successor

If K has two extensions L and R, $L(B) = T$ and $R(B) = F$, and Q_α is its right extension, i.e. R, we show by the next lemma that Q_α cannot belong to the quasi-rightmost path.

Lemma 36 *Q_α is not a node of the quasi-rightmost path.*

Proof: *Q_α being the right extension of K, $Q_\alpha(B) = F$ and $K(A_\beta) = T$. We have to study two cases :*

- **If A_β is K-reducible** *by an equation $(g \simeq d)$ into $(u_1[d] \simeq v)$: g and d are not AC-equal by minimality of A_β. So, $g \succ d$ but in his case Q_α should be a quasi-failure node.*
- **If A_β is K-irreducible**, *the right node R_1 of A_β is either a failure node, or a distant failure node, or a quasi-failure node.*
  1. *R_1 cannot be a failure node, because it would be labeled by a clause $(u_1 \simeq v) \vee D_1$ of $\mathcal{GS}$ and the AC-equal clause $(u_2 \simeq v) \vee D_1$ would belong to $\mathcal{GS}$ too and falsified by Q_α. This interpretation should not be in $MCT(\mathcal{S}^*)$.*
  2. *R_1 cannot be a distant failure node, because it would be labeled by a clause $(u_1 \simeq w) \vee D_1$ of $\mathcal{GS}$ and Q_α would be a distant failure node too, labeled by $(u_1 \simeq w) \vee D_1$, by definition of distant failure nodes.*
  3. *R_1 cannot be a quasi-failure node, because it would mean that Q_α should be a quasi-failure node too.*

We have so proved that Q_α could not be a node of the quasi-rightmost path. □

4.3 Proof of the Completeness Theorem

We have proved that if the empty clause is not generated, the quasi-rightmost path of $MCT(\mathcal{S}^*)$ is non-empty and AC-consistent. We are going to show that the last node of this quasi-rightmost path Q_γ falsifies a clause of $\mathcal{GS}$, which will imply that $MCT(\mathcal{S}^*)$ is empty, and also that the empty clause has been generated.

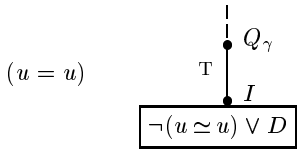
Proposition 37 *The quasi-rightmost path is empty, and also the empty clause belongs to S^*.*

Proof: *The last node Q_γ of the quasi-rightmost path is defined on W_γ, i.e. on all atoms A_i for $i < \gamma$; Q_γ belongs to $MCT(\mathcal{S}^*)$ and its extensions are failure, distant failure or quasi-failure nodes. Let us denote B the atom A_γ. As in the empty theory, there are three main cases to study :*

1. *Either Q_γ has a single extension I and B is Q_γ-irreducible,*
2. *Or Q_γ has two extensions L and R (B is Q_γ-irreducible),*
3. *Or Q_γ has a single extension I and B is Q_γ-reducible.*

Let us detail these cases :

Case 1 : *Q_γ has a single extension I and B is Q_γ-irreducible*

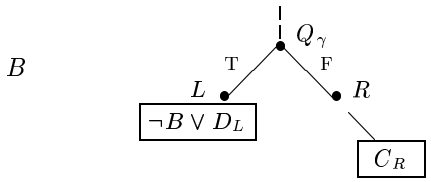


By construction of the semantic tree, I satisfies the atom B, which is an equation $(u \simeq u)$ where u is a ground irreducible term. But, since I is a failure node, it falsifies a clause C of $\mathcal{S}^*$.
Q_γ not falsifying C and Q_γ and I differing only by their value on B, we have

$$C \;=\; \neg(u \simeq u) \vee D \quad \text{and} \quad Q_\gamma(D) = F \;,$$

for a clause D whose atoms are smaller than $(u \simeq u)$.
The clause C belongs to $\mathcal{GS}$ and there is a clause C' in S^* and a ground substitution σ such that $C'\sigma =_{AC} C$. An AC-reflexion in the clause $C'\sigma$ produces a clause AC-equal to D ; so, D belongs to $\mathcal{GS}$, and since it is falsified by Q_γ, this last interpretation cannot belong to $MCT(\mathcal{S}^*)$, and therefore to the quasi-rightmost path.
This first case is also impossible.

Case 2 : *Q_γ has two extensions L and R (B is Q_γ-irreducible)*



The node Q_γ has two extensions which are failure, distant failure or quasi-failure nodes. Its left extension L is also a failure node, labeled by a clause $\neg B \vee D_L$ of $\mathcal{GS}$, where each atom of D_L is smaller than B. Since Q_γ and L differ only by their value for B, Q_γ falsifies D_L. Let us denote C'_L the clause $\neg B' \vee D'_L$ of S^* fulfilling $C_L =_{AC} C'_L\sigma$, for a ground substitution σ.
About R, this is either a failure, or a distant failure, or a quasi-failure node. Let us study these cases.

Case 2.1 : *If R is a failure node*

The clause C_R labeling R may belong to $\mathcal{GS}$ or $\mathcal{AC}$.

Case 2.1.1 : *If $C_R \in \mathcal{GS}$*

Then, C_R is a clause $B \vee D_R$ and there is a clause C'_R in S^* such that $C'_R\sigma =_{AC} C_R$. As for D_L, Q_γ falsifies D_R.
Hence, an AC-resolution step between $C'_L\sigma$ and $C'_R\sigma$ produces a clause AC-equal to $D_L \vee D_R$; this clause belongs to $\mathcal{GS}$ and, since it is falsified by Q_γ, the interpretation Q_γ cannot be a node of $MCT(\mathcal{S}^*)$.

Case 2.1.2 : *If $C_R \in \mathcal{AC}$*

It means that B is an equation $(u \simeq v)$ where u and v are AC-equal. So, an AC-reflexion in the clause $C'_L\sigma$ produces the clause $D'_L\sigma$, AC-equal to D_L. D_L is also a clause of $\mathcal{GS}$ and Q_γ cannot be a node of $MCT(\mathcal{S}^*)$ since it falsifies D_L.

Case 2.2 : *If R is a quasi-failure node*

By definition of a quasi-failure node, B is an atom $(u_2 \simeq v)$ where $u_2 \succ v$, and R is labeled by an equation $(u_2 \simeq u_1[g])$ where u_2 and $u_1[g]$ are AC-equal and $(u_1[g] \simeq v)$ is Q_γ-reducible by an equation $(g \simeq d)$ $(g \succ d)$ into $(u_1[d] \simeq v)$; $Q_\gamma(u_1[d] \simeq v) = T$.

Proposition 27 permits to say that there is an equation $(l \simeq r)$ whose right node is a failure or distant failure node and which reduces an atom AC-equal to $(u_1[g] \simeq v)$, i.e. AC-equal to B.

Let C_1 be the clause labeling the right node of $(l \simeq r)$. By Proposition 28, an inference step (superposition or contextual superposition) from C_1 into C_L produces a clause falsified by Q_γ. So, Q_γ cannot belong to $MCT(\mathcal{S}^*)$.

Case 2.3 : *If R is a distant failure node*

Then, B is an atom $(u \simeq v)$ and a failure node M_R below R occurs at the level of an atom $(u \simeq w)$ and is labeled by a clause $(u \simeq w) \vee D_R$; moreover, $Q_\alpha(w \simeq v) = T$. Let C'_R be the clause of S^* such that $C'_R\sigma =_{AC} (u \simeq w) \vee D_R$. If we apply a superposition from $C'_R\sigma$ into $C'_L\sigma$, the deduced clause is AC-equal to $\neg(w \simeq v) \vee D_L \vee D_R$. This last clause belongs to $\mathcal{GS}$ and is falsified by M_R. It means that a failure node should occur above M_R, which is in contradiction with the construction of the quasi-rightmost path.

So, case 2 cannot hold.

Case 3 : *Q_γ has a single extension I and B is Q_γ-reducible.*

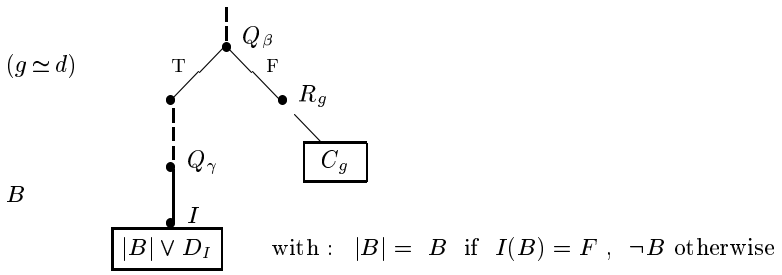
The interpretation I is a failure node, since B is reducible. Let C_I be the clause labeling I. It cannot belong to $\mathcal{AC}$ because B would be an equation $(s \simeq s')$ with $s =_{AC} s'$, reducible to $(s[d] \simeq s')$ by an atom $(g \simeq d)$; but, this last atom, falsified by Q_γ, is AC-equal to $(s[d] \simeq s)$, which is valid for Q_γ since it is reducible to $(s[d] \simeq s[d])$; it contradicts the AC-consistency of Q_γ.

Thus, C_I is a clause $|B| \vee D_I$ of $\mathcal{GS}$, where $|B|$ denotes the literal B if $I(B) = F$, or $\neg B$ if $I(B) = T$.

We can note that B cannot be reduced into an AC-equal atom B', since the clause $|B'| \vee D_I$ is in $\mathcal{GS}$ too, and should also label a failure node at the level of B'.

By Proposition 27, there exist two atoms B_2 and $(g \simeq d)$ such that B_2 is AC-equal to B, reducible by $(g \simeq d)$ with $g \succ d$, and $I(B) = I(B_2[d])$. Let β be the index of the atom $(g \simeq d)$ and Q_β the restriction of Q_γ to W_β (defined on atoms smaller than $(g \simeq d)$). The atom $(g \simeq d)$ is Q_β-irreducible and its right node is a failure node labeled by $(g \simeq d) \vee D_g$, or a distant failure node labeled by $(g \simeq w) \vee D_g$.

At last, if B is an equation $(s \simeq t)$ valid for I, where $s > t$, we will consider that the equation $(g \simeq d)$ reduces $B_2 = (s_2 \simeq t_2)$ into s_2, i.e. in the largest side of the equality. This is justified by the fact that s_2 has to be reducible : if t_2 can be reduced into t'_2, $(s_2 \simeq t_2)$ is reducible by $(s_2 \simeq t'_2)$.



Let us detail the possible cases :

Case 3.1 : *If $I(B) = T$ or B is not equational*

If B fulfills one of these conditions, $|B|$ is the maximal literal w.r.t. $\succ_L$ of the clause $|B| \vee D_I$. Hence, Proposition 28 says that a superposition or contextual superposition from C_g into C_I produces a clause falsified by Q_γ or M_g (M_g is the failure node at or below R_g). It contradicts the construction of the quasi-rightmost path.

Case 3.2 : *If $I(B) = F$ and $B = (s \simeq t)$*

The clause C_I is $(s \simeq t) \vee D_I$. In this case, the maximality w.r.t. $\succ_L$ of B in C_I is not guaranty. In fact, if there is a literal $\neg(s' \simeq u)$ in D_I such that $s' =_{AC} s$, we cannot apply

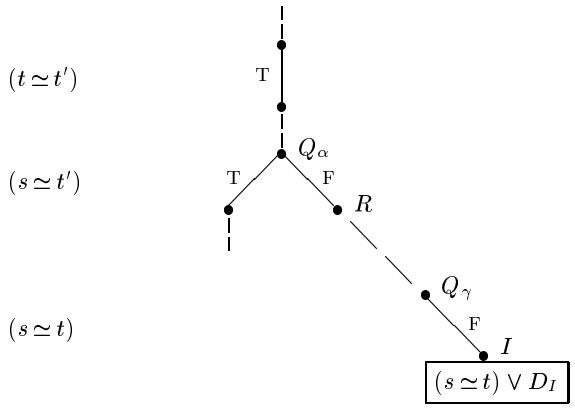
any superposition into B. This situation takes place only if $(s \simeq t)$ is reducible at a position in s. So, let us study the reducibility of s :

Case 3.2.1 : *If s is reducible*

We just saw that B may be non-maximal in C_I. Let us assume that C_I can be decomposed into $(s \simeq t) \vee \neg(s \simeq t_1) \vee \ldots \vee \neg(s \simeq t_n) \vee D'_I$. Q_γ satisfies atoms $(s \simeq t_i)$ and falsifies D_I. Atom $(s \simeq t_i)$ contain s and not a term s' AC-equal to s because, by AC-consistency, $Q_\gamma(s' \simeq t_i) = Q_\gamma(s \simeq t_i)$.
Then, we can apply a sequence of (contextual) superpositions from C_g into each s of literals $\neg(s \simeq t_i)$ of C_I. These literals are transformed into $\neg(s_2[d] \simeq t_i)$ if R_g is a failure node, or else into $\neg(s_2[w] \simeq t_i)$ if R_g is a distant failure node at the level of $(g \simeq w)$. For simplifying, let us denote these literals $\neg(s'_2 \simeq t_i)$. The deduced clause is $(s \simeq t) \vee \neg(s'_2 \simeq t_1) \vee \ldots \vee \neg(s'_2 \simeq t_n) \vee D'_I \vee D_g$. It no more contains negative equational literals whose largest side is AC-equal to s, and I falsifies each literal $\neg(s'_2 \simeq t_i)$.
We have just shown that I falsifies a clause of $\mathcal{GS}$ in which $(s \simeq t)$ is maximal w.r.t. $\succ_L$.
However, if R_g is a distant failure node, an atom $(g \simeq v)$ of D_g may be greater than $(s \simeq t)$ et and also out of the domain of I (in this case, g and s are AC-equals). But, then M_g, failure node below R_g, falsifies the deduced clause. In each case, there is a contradiction with the construction of the quasi-rightmost path.
From now on, we can consider that $(s \simeq t)$ is maximal w.r.t. $\succ_L$ in C_I. The last condition to check for applying a (contextual) superposition from C_g into C_I is that $(g \simeq d)$ (or $(g \simeq w)$) is smaller than $(s \simeq t)$.
If $(g \simeq w) \succ (s \simeq t)$, g and s are AC-equal and R_g is a distant failure node with M_g as failure node at the end of its rightmost path (at the level of $(g \simeq w)$) ; but, M_g falsifies the clause $(g \simeq t) \vee D_I$, AC-equal to C_I, because by definition $M_g(g \simeq v) = F$ for each term $v < w$. It contradicts the construction of the quasi-rightmost path since by definition there is no failure node between R_g and M_g.
Now that all conditions are fulfilled, we can apply a (contextual) superposition from C_g into C_I and deduce the clause $(s'_2 \simeq t) \vee D_I \vee D_g$. This clause is falsified by Q_γ and should label a failure node cutting the quasi-rightmost path.

Case 3.2.2 : *If s is irreducible*

There is no negative literal $\neg(s' \simeq u)$ in D_I such that s' is AC-equal to s. It comes from the fact that if such a literal existed, $(s \simeq u)$ should be valid for Q_γ too (by AC-consistency) and also $(s \simeq t)$ should be reducible by this atom $(s \simeq u)$ into $(u \simeq t)$. This is impossible since s is irreducible.
Let t' be the smallest term less than t and such that $Q_\gamma(t \simeq t') = T$ and $(s \simeq t')$ is irreducible. The atom $(s \simeq t)$ is reducible by $(t \simeq t')$ into $(s \simeq t')$, and $(s \simeq t')$ is falsified by Q_γ. Let α be the index of $(s \simeq t')$; Q_α is the restriction of Q_γ to W_α. Let R be the right extension of Q_α : $R(s \simeq t') = F$.
The situation is depicted in next figure.



The node R is like a distant failure node. To confirm this, we only have to check the following property : $\forall u < t,\ I(s \simeq u) = F$. But, no term u can contradict this, else $(s \simeq t)$ would be reducible in s by this equation $(s \simeq u)$, which is impossible thanks to the hypothesis of irreducibility of s.

Thus, each subcase of case 3 yields a contradiction.

As each case yields an incoherence with the construction of the quasi-rightmost path Q_γ, we conclude that this path is empty, and also that the subtree $MCT(\mathcal{S}^*)$ is empty too. But, the only means to keep just the empty interpretation in $MCT(\mathcal{S}^*)$ are that the empty clause belongs to S^*.

This ends our completeness proof. □

4.4 Compatibility with Simplification

To show that previous proof is compatible with simplification, we only need to prove that failure nodes are labeled by persistent clauses. In this purpose, we defined the following notations :

$GR(S)$: set of clauses AC-equal to a ground instance of a clause of S, i.e.
$\{\ C \mid \exists C' \in S\ ,\ C =_{AC} C'\sigma$, for a ground substitution $\sigma\ \}$

and for a failure node K :

Σ_K	:	set of clauses of S^* falsified by K, i.e. $\{\ C \in S^* \mid \exists C' \in GR(\{C\})\ ,\ K(C') = F\ \}$
TG_K	:	set of minimal clauses of $GR(\Sigma_K)$ w.r.t. $\succ$
Π_K	:	subset of clauses of Σ_K having an instance in TG_K (modulo AC)
Π'_K	:	set of minimal clauses of Π_K w.r.t. strict subsumption

We can note that $GR(\Pi_K) = GR(\Pi'_K) = TG_K$, and $\mathcal{GS} = GR(S^*)$.

In the next proposition, we show that each failure node falsifies a persistent clause. This includes distant failure nodes, since they are built from a failure node.

Proposition 38 *Let K be a failure node, extension of a node Q_α of the quasi-rightmost path. If Q_α is AC-consistent, K falsifies a persistent clause.*

Proof: *By induction hypothesis, let us assume that every failure node along the quasi-rightmost path (above K) is labeled by a persistent clause. For proving that K falsifies a persistent clause, we are going to show that each element of Π'_K is persistent. Let C' be a clause of Π'_K and C a clause of TG_K which is AC-equal to a ground instance $C'\sigma$ of C' ; K falsifies the clause C. C' is persistent if it is never simplified or subsumed :*

If C' is simplified : *There is an index j for which C' belongs to S_j and a clause $(l' \simeq r') \vee D'$ of S_j simplifies C' at a non-variable position p of a literal L', using a substitution ρ. By definition of simplification rules, the clause C' can be decomposed into $L' \vee C'_1 \vee C'_2$, where $D'\rho$ is included into C'_2 (modulo AC), i.e. $D'\rho \subseteq_{AC} C'_2$. In the same way, the clause C can be decomposed into $L \vee C_1 \vee C_2$, where $L =_{AC} L'\sigma$, $C_1 =_{AC} C'_1\sigma$ and $D'\rho\sigma \subseteq_{AC} C_2 =_{AC} C'_2\sigma$.*

Moreover, there is an atom A in $L' \vee C'_1$ which is greater than $(l'\rho \simeq r'\rho)$ w.r.t. $\succ$. By stability of $\succ$, $A\sigma$ is greater than $(l\rho\sigma \simeq r\rho\sigma)$, and the clause $(l'\rho\sigma \simeq r'\rho\sigma) \vee C_2$ is smaller than C. But, C being in TG_K, K cannot falsify a clause smaller than C, and as K falsifies C_2, K has to satisfy $(l'\rho\sigma \simeq r'\rho\sigma)$.
Another important point is that, $L'_{|_p}$ being AC-equal to $l'\rho$ or $f(l', z)\rho$, there is a literal $L''[l'\rho\sigma]$ AC-equal to $L'\sigma$ and L.

If Q_α falsifies the literal $L''[r'\rho\sigma]$, the clause $L''[r'\rho\sigma] \vee C_1 \vee C_2$ is falsified by K and is smaller than C. It is AC-equal to the ground instance $L'[r'\rho]_p\sigma \vee C'_1\sigma \vee C'_2\sigma$ (or $L'[f(r', z)\rho]_p\sigma \vee C'_1\sigma \vee C'_2\sigma$) of the clause deduced by the simplification step. So, C cannot belong to TG_K.

If the literal $L''[r'\rho\sigma]$ is valid for Q_α, L is the maximal literal in C by AC-consistency of Q_α and because K falsifies L. We can note that K cannot be the failure node below a distant failure node, because L and $L''[r'\rho\sigma]$ would be atoms $(u \simeq w)$ and $(u'' \simeq w''[r'\rho\sigma])$, where $u =_{AC} u''$ and $w =_{AC} w''[r'\rho\sigma]$; then, by definition of a distant failure node, $K(u \simeq w''[r'\rho\sigma]) = F$; the clause $(u \simeq w''[r'\rho\sigma]) \vee C_1 \vee C_2$, AC-equal to the clause deduced by the simplification step, would be falsified by K and, as it is smaller than C, the clause C should not be in TG_K.
K is therefore a direct extension of Q_α and L is either A_α or $\neg A_\alpha$.

- If Q_α has two extensions :
  - If K is the right extension of Q_α : $K(A_\alpha) = F$ and $L = A_\alpha$. *K should be a quasi-failure node and not a failure node, because $L''[l'\rho\sigma]$, AC-equal to L, is reducible by $(l'\rho\sigma \simeq r'\rho\sigma)$.*
  - If K is the left extension of Q_α : $K(A_\alpha) = T$ and $L = \neg A_\alpha$. *The right node R of Q_α is a failure, a distant failure or a quasi-failure node.*
    - If R is a failure node : *By Proposition 27, there exists an irreducible equation $(g \simeq d)$ whose right node is a failure or a distant failure node, and which reduces an atom AC-equal to A_α. A (contextual) superposition can also be applied from the clause labeling the right node of $(g \simeq d)$ into the clause labeling R. Both clauses are persistent by induction hypothesis ; the deduced clause either belongs to $\mathcal{S}^*$ or is subsumed by a clause of $\mathcal{S}^*$; in both cases, a clause of $\mathcal{S}^*$ is falsified by Q_α and this contradicts the construction of the quasi-rightmost path. This case is also impossible.*
    - If R is a quasi-failure node : *There exists an atom A_β AC-equal to A_α and reducible by an equation $(g \simeq d)$, where $g \succ d$, and such that $Q_\alpha(A_\beta[d]) = T$.*
      Let $A_{\beta'}$ be the atom corresponding to the literal L'' ; we know that it is reducible by the equation $(l'\rho\sigma \simeq r'\rho\sigma)$ and $Q_\alpha(A_{\beta'}[r'\rho\sigma]) = F$.
      By Lemma 31, Q_α falsifies a clause of $\mathcal{GS}$, produced by an extended superposition between clauses labeling failure or distant failure nodes along the quasi-rightmost path. The clauses are persistent by induction hypothesis, so the deduction is always possible.
    - If R is a distant failure node : *A_α is an equation $(u \simeq v)$ and the atom corresponding to L'' is $\neg(u'' \simeq v'')$, where $u'' =_{AC} u$ and $v'' =_{AC} v$. $(u'' \simeq v'')$ is reducible by $(l'\rho\sigma \simeq r'\rho\sigma)$.*
      If the reduction applies at a position in v'', the atom $(v \simeq v'')$ being valid for Q_α by AC-consistency, $(v \simeq v''[r'\rho\sigma])$ is valid for Q_α too. But, this last equation should reduce $(u \simeq v)$, which is impossible since A_α is irreducible.
      Thus, $(u'' \simeq v'')$ is reducible at a position in u''. Let M_R be the failure node below R, labeled by a clause $C_M = (u \simeq w) \vee D_M$. By induction hypothesis, this clause is persistent and, by Proposition 27, there exists an atom $(g \simeq d)$ reducing an atom AC-equal to $(u \simeq w)$ and whose right node is a failure or a distant failure node labeled by a clause C_g of $\mathcal{GS}$; C_g is persistent too. The clause deduced by a (contextual)

superposition from C_g into C_M is falsified by M, smaller than C_M, and belongs to $\mathcal{S}^*$; it should also label a failure node above M ; it contradicts the construction of the quasi-rightmost path.

Each case yields a contradiction.

- If Q_α has a single extension (K) :
  - If A_α is Q_α-irreducible : *A_α is an equation $(u \simeq u)$ and $K(u \simeq u) = T$; L is $\neg(u \simeq u)$. Let A_β be the atom corresponding to L'', $L'' = \neg A_\beta$. A_β is reducible by $(l'\rho\sigma \simeq r'\rho\sigma)$ and $A_\beta[r'\rho\sigma]$ is AC-equal to an equation $(u''[l'\rho\sigma] \simeq u''[r'\rho\sigma])$, where $u''[l'\rho\sigma] =_{AC} u$. By AC-consistency of Q_α, $Q_\alpha(A_\beta[r'\rho\sigma]) = Q_\alpha(u''[l'\rho\sigma] \simeq u''[r'\rho\sigma]) = F$. But, this last atom is reducible to $(u''[r'\rho\sigma] \simeq u''[r'\rho\sigma])$, which should be falsified by Q_α by definition of reduction and satisfied by Q_α by AC-consistency. So, we have found a contradiction.*
  - If A_α is Q_α-reducible *by an equation $(g \simeq d)$, with $g \succ d$ (g and d cannot be AC-equal, otherwise there should be a failure node above K), and $Q_\alpha(L[d]) = F$. We know that L'' is reducible by $(l'\rho\sigma \simeq r'\rho\sigma)$ and $Q_\alpha(L''[r'\rho\sigma]) = T$.*
    As L and L'' are AC-equal, Lemma 31 says that Q_α falsifies a clause of $\mathcal{GS}$.

Each case is impossible. So, C' cannot be simplified.

If C' is strictly subsumed : *There exists an index j for which C' belongs to S_j and a clause D' of S_j strictly subsumes C'. It means there is a substitution ρ such that each literal of $D'\rho$ is AC-equal to a literal of C'.*
The clause C can be decomposed into two multisets of literals : $C = C_1 \vee C_2$, where C_2 is AC-equal to $D'\rho\sigma$ (C_1 may be empty). As K falsifies C_2, D' belongs to Σ_K and either $C_2 \prec C$ and C cannot be in TG_K, or D' belongs to Π_K and C' cannot be in Π'_K since it is subsumed by D'.
The conclusion is : if a clause is strictly subsumed, it cannot belong to Π'_K.

As the clause C' is neither simplified nor subsumed, it is persistent. □

We have proved that failure nodes were labeled by persistent clauses. The proof was done by induction on nodes of the quasi-rightmost path, assuming at each step that the handled part of this path was AC-consistent. So, there is no problem with the proof of AC-consistency of this path.

5 Further Works and Conclusion

We have designed a refutationally complete superposition-based inference system for associative and commutative deduction. It was adapted from the ordered AC-paramodulation strategy of [RV93]. In addition, we have shown that our inference rules were compatible with simplification. This is a fundamental issue for the efficiency of theorem-proving. Our proof of completeness is based on the transfinite semantic tree technique, and we have shown that there is no need to weaken the superposition strategy as it was done in earlier works with semantic trees.

Such strategies of AC-paramodulation or AC-superposition were recently adapted by including symbolic constraints in order to never compute AC-unifiers [Vig93, NR94].

This idea of replacing axioms by ad-hoc mechanisms such as unification algorithm, inference rules or symbolic constraints can be further extended to other equational or non equational theories. In general the efficiency gain is noticeable, but this still needs to be carefully studied by experimentations.

References

[AHM89] S. Anantharaman, J. Hsiang, and J. Mzali. Sbreve2: A term rewriting laboratory with (AC-)unfailing completion. In N. Dershowitz, editor, *Proceedings 3rd Conference on Rewriting Techniques and Applications, Chapel Hill (N.C., USA)*, volume 355 of *Lecture Notes in Computer Science*, pages 533–537. Springer-Verlag, April 1989.

[BG90] L. Bachmair and H. Ganzinger. On restrictions of ordered paramodulation with simplification. In M. E. Stickel, editor, *Proceedings 10th International Conference on Automated Deduction, Kaiserslautern (Germany)*, volume 449 of *Lecture Notes in Computer Science*, pages 427–441. Springer-Verlag, July 1990.

[BG93] L. Bachmair and H. Ganzinger. Associative-Commutative Superposition. Technical report MPI-I-93-267, Max Planck Institut für Informatik, Saarbrücken, 1993.

[BHK+88] H.-J. Bürckert, A. Herold, D. Kapur, J. Siekmann, M. E. Stickel, M. Tepp, and H. Zhang. Opening the AC-unification race. *Journal of Automated Reasoning*, 4(1):465–474, 1988.

[DJ90] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*. Elsevier Science Publishers B. V. (North-Holland), 1990.

[Dom91] E. Domenjoud. *Outils pour la déduction automatique dans les théories associatives-commutatives*. Thèse de Doctorat d'Université, Université de Nancy 1, September 1991.

[Hab90] M. Haberstrau. About the completeness of strict superposition. Internal report 156 RR 595, Univ. de Paris Sud, Paris 11. Orsay, 1990.

[HR91] J. Hsiang and M. Rusinowitch. Proving refutational completeness of theorem proving strategies: The transfinite semantic tree method. *Journal of the Association for Computing Machinery*, 38(3):559–587, July 1991.

[JK86] J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computing*, 15(4):1155–1194, 1986. Preliminary version in Proceedings 11th ACM Symposium on Principles of Programming Languages, Salt Lake City (USA), 1984.

[JK91] J.-P. Jouannaud and C. Kirchner. Solving equations in abstract algebras: a rule-based survey of unification. In J.-L. Lassez and G. Plotkin, editors, *Computational Logic. Essays in honor of Alan Robinson*, chapter 8, pages 257–321. MIT Press, Cambridge (MA, USA), 1991.

[KB70] D. E. Knuth and P. B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, Oxford, 1970.

[Lai89] M. Lai. On how to move mountains 'associatively and commutatively'. In N. Dershowitz, editor, *Proceedings 3rd Conference on Rewriting Techniques and Applications, Chapel Hill (N.C., USA)*, volume 355 of *Lecture Notes in Computer Science*, pages 187–202. Springer-Verlag, April 1989.

[Lan79] D. S. Lankford. Mechanical theorem proving in field theory. Technical report, Louisiana Tech. University, 1979.

[LB77] D. S. Lankford and A. Ballantyne. Decision procedures for simple equational theories with associative commutative axioms: complete sets of associative commutative reductions. Technical report, Univ. of Texas at Austin, Dept. of Mathematics and Computer Science, 1977.

[Lov78] D. Loveland. *Automatic Theorem Proving*. Elsevier Science Publishers B. V. (North-Holland), 1978.

[NR91] P. Narendran and M. Rusinowitch. Any ground associative-commutative theory has a finite canonical system. In R. V. Book, editor, *Proceedings 4th Conference on Rewriting Techniques and Applications, Como (Italy)*. Springer-Verlag, 1991.

[NR94] R. Nieuwenhuis and A. Rubio. AC-superposition with constraints: no AC-unifiers needed. Technical report, Technical University of Catalonia, Barcelona, 1994. To appear in Proc. of CADE 12.

[Pet83] G. Peterson. A technique for establishing completeness results in theorem proving with equality. *SIAM Journal of Computing*, 12(1):82–100, 1983.

[Pet91] U. Petermann. Building in equational theories into the connection method. In P. Jorrand and J. Kelemen, editors, *Fundamental of Artificial Intelligence Research*, volume 535 of *Lecture Notes in Computer Science*, pages 156–169. Springer-Verlag, 1991.

[Plo72] G. Plotkin. Building-in equational theories. *Machine Intelligence*, 7:73–90, 1972.

[PS81] G. Peterson and M. E. Stickel. Complete sets of reductions for some equational theories. *Journal of the Association for Computing Machinery*, 28:233–264, 1981.

[RN93] A. Rubio and R. Nieuwenhuis. A precedence-based total ac-compatible ordering. In C. Kirchner, editor, *Proceedings 5th Conference on Rewriting Techniques and Applications, Montreal (Canada)*, volume 690 of *Lecture Notes in Computer Science*, pages 374–388. Springer-Verlag, 1993.

[Rus89] M. Rusinowitch. *Démonstration automatique-Techniques de réécriture*. InterEditions, 1989.

[Rus91] M. Rusinowitch. Theorem-proving with resolution and superposition. *Journal of Symbolic Computation*, 11:21–49, 1991.

[RV91] M. Rusinowitch and L. Vigneron. Automated deduction with associative commutative operators. In P. Jorrand and J. Kelemen, editors, *Fundamental of Artificial Intelligence Research*, volume 535 of *Lecture Notes in Computer Science*, pages 185–199. Springer-Verlag, 1991.

[RV93] M. Rusinowitch and L. Vigneron. Automated deduction with associative commutative operators. Internal report 1896, Institut National de Recherche en Informatique et Automatique, May 1993. To appear in *J. of Applicable Algebra in Engineering, Communication and Computation*.

[RW69] G. A. Robinson and L. T. Wos. Paramodulation and first-order theorem proving. In B. Meltzer and D. Mitchie, editors, *Machine Intelligence 4*, pages 135–150. Edinburgh University Press, 1969.

[Sti81] M. E. Stickel. A unification algorithm for associative-commutative functions. *Journal of the Association for Computing Machinery*, 28:423–434, 1981.

[Sti84] M. E. Stickel. A case study of theorem proving by the Knuth-Bendix method: Discovering that $x^3 = x$ implies ring commutativity. In R. Shostak, editor, *Proceedings 7th International Conference on Automated Deduction, Napa Valley (Calif., USA)*, volume 170 of *Lecture Notes in Computer Science*, pages 248–258. Springer-Verlag, 1984.

[Vig93] L. Vigneron. Associative-commutative deduction with constraints. Technical Report 93-R-196, CRIN, 1993. To appear in Proc. of CADE 12.

[Wer92] U. Wertz. First-order theorem proving modulo equations. Technical Report MPI-I-92-216, Max Planck Institut für Informatik, April 1992.