

Reduction and Narrowing for Horn Clause Theories

P. PADAWITZ

Universitat Passau, Postfach 2540, D-8390 Passau, West Germany

We give an overview of rewrite rule based calculi for reasoning about Horn clause specifications. Such reasoning consists in computing (normal forms), proving (theorems) or solving (goals). Computations are performed by term reduction, proofs are carried out by goal reduction, solutions are obtained by narrowing. The presentation follows our book 'Computing in Horn Clause Theories' where we took the viewpoint that most results in this area centre around the soundness or completeness of an inference system. Besides the general calculi of reduction and narrowing we discuss specialisations, which embody particular strategies. For applications to data type specification and program verification, we present the foundations of inductive proof methods based on reduction and narrowing.

Received August 1990

INTRODUCTION

Let us first recall the case where all axioms are equations. Given a signature SIG and a set of E equations, the *term reduction relation* \rightarrow_E is the least reflexive, transitive and SIG-compatible (monotone) closure of the set of *instances* of equations of E. (The transitive and monotone, but non-reflexive, closure will be denoted by \rightarrow_E^+ .) If two terms t and t' satisfy $t \rightarrow_E t'$, one says that t *rewrites into t' via E*. The original purpose of term reductions is to solve the *word problem of E*: Under which conditions on E is the set of *valid* equations decidable? Validity means, from the semantical viewpoint, validity in all (SIG-) models of E, or, equivalently, from the proof-theoretical viewpoint, derivability from E via the inference rules of the equational calculus (cf. Birkhoff & Moore).² Let us call an equation $t \equiv t'$ an *E-theorem* if it is derivable from E by applying these rules. (\equiv stands for an equality predicate *symbol*, in contrast to $=$, which expresses identity on the meta-level of reasoning about formulas.)

The difference between a reduction $t \rightarrow_E t'$ and an E-theorem $t \equiv t'$ is the symmetry rule involved in the equational calculus, but excluded from \rightarrow_E . The core of rewriting theory lies in proving $t \equiv t'$ as an E-theorem by rewriting t and t' into a common term u : $t \rightarrow_E u$ and $t' \rightarrow_E u$. E is called *Church-Rosser* if every E-theorem $t \equiv t'$ can be proved in this way. If, in addition, \rightarrow_E^+ is a well-founded relation, the validity of $t \equiv t'$ becomes decidable: Construct reductions $t \rightarrow_E u$ and $t \rightarrow_E u'$ such that u and u' are *E-normal forms*, i.e., if $u \rightarrow_E v$ and $u' \rightarrow_E v'$, then $u = v$ and $u' = v'$; and check whether u and u' are identical. If they are, $t \equiv t'$ is an E-theorem, otherwise it is not.

In fact, the Church-Rosser property can be expressed in a more abstract way. Instead of regarding the *terms* t and t' as the basic units to be modified we may view reductions, analogously to inferences via the equational calculus, as transformations of *equations*, constituting the inference relation \vdash_E of *goal reduction*. So we combine $t \rightarrow_E u$ and $t \rightarrow_E u'$ (see above) into a goal reduction $t \equiv t' \vdash_E u \equiv u'$. Given that $t \equiv t'$ is the theorem to be proved, derivations via the equational calculus start out from axioms of E and try to achieve $t \equiv t'$, whereas goal reduction proceeds from $t \equiv t'$ by term replacement down to a reflexive equation of the

form $u \equiv u$. So \vdash_E is generated by two inference rules on equations. The first defines single reduction steps:

(RED) For all equations e , variables x , $u \equiv u' \in E$ and substitutions f , $e[u[f]/x] \vdash_E e[u'/f]/x]$.

Here $u[f]$ denotes the instance of u by the substitution f . Moreover, x is a variable occurring in the equation e , and $e[u[f]/x]$ denotes the equation obtained from e by substituting $u[f]$ for x .

The second rule closes a sequence of reduction steps successfully and hence is called the *success rule*:

(SUC) For all terms t , $t \equiv t \vdash_E \emptyset$.

Using goal reduction we may define: E is Church-Rosser if $t \equiv t' \vdash_E \emptyset$ whenever $t \equiv t'$ is derivable from E via the equational calculus. Hence saying that E is Church-Rosser is another way of stating that the calculus of goal reduction, given by RED and SUC, is *complete* with respect to the equational calculus and thus with respect to validity in the class of all models of E. An important consequence of the Church-Rosser property is the uniqueness of E-normal forms: If $t \equiv t'$ is derivable from E via the equational calculus, but t and t' are irreducible with respect to \rightarrow_E , then t and t' must be equal.

This concept will be generalised in several respects. First, axioms are allowed to be arbitrary Horn clauses. Equality predicates carry on playing a dominant role, but other predicates may occur as well. Second, we will prove by reduction not only equations, but also other (atomic) formulas, as far as the signature includes non-equality predicates. Third, we want to cope with cases where not the entire proof of a theorem can be carried out by goal reduction. In other words, we want to weaken the notion of a *succeeding* goal reduction: why should the reduction process be finished only if a reflexive equation has been obtained? A more general approach would allow us to stop already if, for a suitable set BAX of *base axioms*, a (not necessarily reflexive) BAX-theorem has been achieved.

The fourth extension of classical rewriting theory leads from *proving* equations to *solving* equations. If goal reduction is extended to a rule that transforms pairs consisting of a goal and a substitution, one comes up with the *narrowing* rule for computing substitutions that validate a given goal. Hence narrowing generalises goal

reduction. From the viewpoint of automated theorem-proving, however, where the basic rule of term replacement is *paramodulation*, narrowing is a specialisation: Given an equational axiom, narrowing allows applying it from left to right, but not from right to left.

The paper is organised as follows. Section 1 provides syntactical conventions and states a general assumption on all specifications we are dealing with. Section 2 is about the notions of theory and model of a Horn clause specification. The *cut calculus with equality* serves as a reference for all deductive systems given later. Section 3 presents conditional reduction in three variants: Goal reduction, *conditional* (term) *reduction* and *term reduction*. The *Church–Rosser property* of a set of Horn clauses, its applications and basic criteria are treated in Section 4. Section 5 shows how *reduction strategies* like ‘innermost’, ‘outermost’, etc. specialise goal reduction. *Basic reduction* yields another specialisation. The topic of Section 6 is narrowing, as a calculus both for solving goals and for proving inductive theorems. Section 7 continues Section 5 by proceeding to *narrowing strategies* and sketching the refinements of narrowing called *basic narrowing* and *optimised narrowing*. Section 8 combines approaches to ‘demand-driven’ narrowing into a calculus called *lazy narrowing*.

Emphasis is laid on a unified presentation, which reveals the basic facts, their interrelations and differences. For this purpose, the main issues are expressed in terms of (properties of) inference systems, all handling the same type of formulas. The reader probably expects examples of specifications and concrete derivations using these inference systems. For several reasons, we did not include such running examples (except in Sections 1 and 2). The first is lack of space. The second is that we do not consider them really helpful. Relevant ideas in this field often become more transparent when they are expressed abstractly rather than encoded into special examples. Thirdly, the whole matter has evolved on the ground of very few examples. (This seems to distinguish reduction and narrowing from classical theorem proving.) In fact, assessing a method or result on the basis of *application-oriented* criteria, is, at least at the moment, rather difficult because practical applications or comparisons of different methods under a practical viewpoint are rare. The latter is not least a consequence of the lack of *unifying* approaches.

So we hope the reader is able to get through our presentation without being provided with concrete examples, but, nevertheless, feeling challenged to make practical experiments with the calculi provided.

1. PRELIMINARIES

We start out from a many-sorted signature $SIG = (S, OP, PR)$ consisting of a set S of *sorts* and S^+ -sorted sets OP and PR of *operation* (or *function*) *symbols* and *predicate symbols*, respectively. For all $s \in S$, PR contains an equality symbol \equiv with sort ss .

Given a fixed S -sorted set X of *variables*, $T(SIG)$ denotes the set of *terms* over SIG , while $GT(SIG)$ comprises all *ground*, i.e. variable-free, terms over SIG . $var(t)$ is the set of variables occurring in t .

As usual, the S -sorted functions from X to $T(SIG)$ (or $GT(SIG)$, respectively) are called (*ground*) *substitutions*. The *instance* of a term t by a substitution f , denoted by

$t[f]$, is obtained by replacing the variables of t according to f . If $dom(f)$, the *domain* of f , given by all variables x with $fx \neq x$, is finite, say $dom(f) = [x_1, \dots, x_n]$, we write $t[f x_1/x_1 \dots, f x_n/x_n]$ instead of $t[f]$. f is a *unifier* of two terms t and t' if $t[f] = t'[f]$.

In addition to SIG , a (Horn clause) *specification* (SIG, AX) contains a set AX of axioms each of which is a (Horn) *clause* of the form $p \Leftarrow \gamma$ where the *conclusion* p is an atom (ic formula) over SIG and the *premise* γ is a *goal*, i.e. a finite set of atoms over SIG . A clause with an empty premise such as $p \Leftarrow \emptyset$ is simply written as p . Hence special clauses are *equations* like $t \equiv t'$ and *conditional equations* such as $t \equiv t' \Leftarrow \gamma$. (The predicate symbols occurring in γ need not be equations.)

General Assumption. Throughout the paper, we fix a specification (SIG, AX) . For reasons already mentioned in the introduction we separate from OP a set BOP of *base operations* and from AX a set BAX of *base axioms*. Terms over BOP are called *base terms*. The conditional equations of BAX are assumed to be symmetric, i.e. if $t \equiv t' \Leftarrow \vartheta$ is a base axiom, then $t' \equiv t \Leftarrow \vartheta$ is a base axiom as well. The remaining clauses of AX are assumed to be conditional equations and are called *rewrite axioms*. RAX denotes the set of these axioms.

The *base signature* $BSIG = (S, BOP, PR)$ is assumed to be *inhabited*, i.e. for all $s \in S$ there is a ground term over $BSIG$ with sort s . $(BSIG, BAX)$ is called the *base specification*. ■

A base specification of sequences and multisets (bags) of natural numbers may read as follows. Equality predicates are not listed explicitly.

SEQ&BAG

sorts	nat, seq, seq ² , bag
	<i>symbol</i> <i>type</i>
opns	0 nat
	$+1$ nat \rightarrow nat
	ε seq
	$_ \& _$ nat, seq \rightarrow seq
	conc($_$, $_$) seq, seq \rightarrow seq
	($_$, $_$) seq, seq \rightarrow seq ²
	\emptyset bag
	add($_$, $_$) nat, bag \rightarrow bag
	seqToBag($_$) seq \rightarrow bag
preds	$_ < _$ nat, nat
	$_ \leq _$ nat, nat
	isSorted($_$) seq
vars	m, n: nat; s: seq; b: bag
axms	conc(ε , s) \equiv s
	conc(n&s, s') \equiv n&conc(s, s')
	add(m, add(n, b)) \equiv add(n, add(m, b))
	seqToBag(ε) $\equiv \emptyset$
	seqToBag(n&s) \equiv add(n, seqToBag(s))
	$0 < n + 1$
	$m + 1 < n + 1 \Leftarrow m < n$
	$m \leq n \Leftarrow m < n$
	$m \leq n \Leftarrow m \equiv n$
	isSorted(ε)
	isSorted(n& ε)
	isSorted(m&n&s) $\Leftarrow m \leq n$, isSorted(n&s)

As an example of (SIG, AX) , let us specify the *quicksort* algorithm.

QUICKSORT

base	SEQ&BAG	
	<i>symbol</i>	<i>type</i>
opns	sort($_$)	$\text{seq} \rightarrow \text{seq}$
	halve($_$)	$\text{seq} \rightarrow \text{seq}^2$
	halveProc($_$)	$\text{seq}, \text{seq}, \text{seq} \rightarrow \text{seq}^2$
	partition($_, _$)	$\text{seq}, \text{nat} \rightarrow \text{seq}^2$
vars	$m, n : \text{nat} ; s, s', s'', \text{left}, \text{right}, \text{less}, \text{greater} : \text{seq}$	
axms	sort(ε) $\equiv \varepsilon$	
	sort($m \& s$) $\equiv \text{conc}(\text{sort}(\text{less}), n \& \text{sort}(\text{greater}))$	
	$\Leftarrow \text{halve}(m \& s) \equiv (\text{left}, n \& \text{right}),$	
	$\text{partition}(\text{conc}(\text{left}, \text{right}), n)$	
	$\equiv (\text{less}, \text{greater})$	
	$\text{halve}(s) \equiv \text{halveProc}(s, \varepsilon, \varepsilon)$	
	$\text{halveProc}(\varepsilon, s, s') \equiv (s, s')$	
	$\text{halveProc}(n \& \varepsilon, s, s') \equiv (s, n \& s')$	
	$\text{halveProc}(m \& n \& s, s', s'') \equiv$	
	$\text{halveProc}(s, m \& s', n \& s'')$	
	$\text{partition}(\varepsilon, n) \equiv (\varepsilon, \varepsilon)$	
	$\text{partition}(m \& s, n) \equiv (m \& \text{less}, \text{greater})$	
	$\Leftarrow m \leq n, \text{partition}(s, n) \equiv (\text{less}, \text{greater})$	
	$\text{partition}(m \& s, n) \equiv (\text{less}, m \& \text{greater})$	
	$\Leftarrow n < m, \text{partition}(s, n) \equiv (\text{less}, \text{greater}).$	

The reader should convince himself that QUICKSORT satisfies the General Assumption.

2. HORN CLAUSE THEOREMS

Definition. The *cut calculus with equality* consists of the congruence axioms for all equality symbols (w.r.t. SIG) and two derivation rules:

- (CUT) $\{p \Leftarrow \gamma \cup \{q\}, q \Leftarrow \delta\} \vdash p \Leftarrow \gamma \cup \delta.$
 (SUB) For all substitutions f , $p \Leftarrow \gamma \vdash p[f] \Leftarrow \gamma[f].$

Let \vdash_c denote the corresponding inference relation. A clause $p \Leftarrow \gamma$ is an *AX-theorem* if $AX \vdash_c p \Leftarrow \gamma$. BAX-theorems are also called *base theorems*. Two terms t and t' are *AX-equivalent* if $t \equiv t'$ is an AX-theorem. Two substitutions f and g are *AX-equivalent* if for all $x \in X$, fx and gx are AX-equivalent. ■

In the case of pure equational specifications (PR being empty and AX being a set of equations), each step in a derivation via the cut calculus with equality corresponds to a step in a derivation via Birkhoff's equational calculus that we mentioned in the introduction.

A *SIG-structure* A interprets each $s \in S$ as a *carrier set* A_s , each $F \in \text{OP}_{ws}$ as a function $F^A : A_w \rightarrow A_s$ and each $P \in \text{PR}_w$ as a relation $P^A \subseteq A_w$. We assume familiarity with the evaluation of terms and the validity of Horn clauses in a SIG-structure. A *SIG-model of AX* is a SIG-structure that satisfies all clauses of AX. Let us denote by $\text{Mod}(AX)$ the class of all SIG-models of AX. The basic completeness result for \vdash_c reads as follows.

Theorem 2.1. (Padawitz,²³ Corollary 4.2.4). $\text{Mod}(\text{SIG}, AX)$ satisfies a clause $p \Leftarrow \gamma$ iff p can be derived from $AX \cup \gamma$ via the cut calculus with equality such that the variables of γ need not be instantiated. ■

For this result, the assumption that BSIG and thus SIG are inhabited is crucial. Otherwise the many-sortedness of SIG could lead to the incorrectness of \vdash_c w.r.t. $\text{Mod}(AX)$ (cf. Goguen).¹¹ For coping with non-inhabited signatures (cf. Section 1),¹¹ proposes a modified calculus where each equation is equipped with a variable

declaration to be taken into account by the inference rules.

In applications to data type specification and program verification, one is rarely interested in the class of *all* SIG-models of AX. Instead, one deals with *term-generated* models, where each carrier element is obtained by evaluating a ground term. Let $\text{Gen}(AX)$ denote this class of models. Proof-theoretically, it is characterised as follows:

Theorem 2.2. (Padawitz, Corollary 4.3.3).²³ $\text{Gen}(AX)$ satisfies a clause $p \Leftarrow \gamma$ iff for all ground substitutions f , $AX \cup \gamma[f] \vdash_c p[f]$. ■

Horn clause specifications admit a *greatest* term-generated model (up to isomorphism), called the *initial* model of AX. Each two of its carrier elements are equal *only if* their term representations are AX-equivalent (see above). With regard to the set of valid ground atoms, the initial model of AX is the *least* one: it satisfies a ground atom p only if p can be derived from AX. The clauses satisfied by the initial model are called *inductive theorems*:

Definition. A clause $p \Leftarrow \gamma$ is an *inductive AX-theorem* if for all ground substitutions f , $AX \vdash_c \gamma[f]$ implies $AX \vdash_c p[f]$. A SIG-model A of AX is *initial* if every SIG-model B of AX admits a unique (SIG-) homomorphism from A to B . $\text{Ini}(AX)$ denotes the (isomorphism) class of initial SIG-models of AX. ■

The correctness of a sorting algorithm like *quicksort* falls into the conditions that (1) the result of applying *quicksort* to a sequence yields a sorted sequence and (2) the sorted sequence is a permutation of the original one. Formally, these conditions hold true if the QUICKSORT-equations (cf. Section 1)

$\text{isSorted}(\text{sort}(s))$ and $\text{seqToBag}(\text{sort}(s)) \equiv \text{seqToBag}(s)$

are inductive AX-theorems where AX is the set of axioms of QUICKSORT.

Theorem 2.3. $\text{Ini}(AX)$ satisfies a clause $p \Leftarrow \gamma$ iff $p \Leftarrow \gamma$ is an inductive AX-theorem. ■

In contrast to the cut calculus, reduction and narrowing lead to *backward* proofs starting from the theorem to be proved and closing with a *reduced* goal. This applies to inductive theorems as well, although we distinguish between two categories of proof methods, namely explicit 'structural' induction on the set of ground substitutions (cf. Burstall,⁴ Boyer & Moore³ and Padawitz²⁴) on the one hand and proofs 'by consistency' on the other hand, where the latter is based upon the following fact.

Theorem 2.4. (Padawitz, Theorem 2.2).²⁵ A clause $p \Leftarrow \gamma$ is an inductive AX-theorem iff $AX \cup \{p \Leftarrow \gamma\}$ is a *conservative extension* of AX, i.e. iff for all ground atoms q , $AX \cup \{p \Leftarrow \gamma\} \vdash_c q$ implies $AX \vdash_c q$. ■

3. CONDITIONAL REDUCTION

The main difference between term rewriting in the equational case and a suitable generalisation to Horn clauses lies in the premises axioms can be equipped with. (The existence of non-equality predicates is not that important, especially as the General Assumption does not admit rewrite axioms for specifying such predicates, unless they were encoded into Boolean functions.) Premises enforce the choice between two possibilities of performing a conditional reduction step. Given a term t

and an axiom $u \equiv u' \Leftarrow \vartheta$ that is applicable to t , i.e. some instance $u[f]$ of u is a subterm of t , one may either

- (1) replace the occurrence of u in t by u' and store the premise ϑ , or
- (2) replace the occurrence of u in t by u' only if $\vartheta[f]$ is valid and forget ϑ after the replacement.

In the second case, one has to make precise the notion of validity used here. Several proposals were given in the literature (cf. e.g. Kaplan,^{19,20} Zhang & Remy,²⁹ Bergstra & Klop¹ and Jouannaud & Waldmann¹⁸). $\vartheta[f]$ may be called valid if it can be derived from AX via the cut calculus (cf. Section 2). Then one actually uses two inference systems, reduction *and* the cut calculus, simultaneously, although the aim is to avoid the forward proofs via CUT and SUB. An alternative is to define validity as *reducibility*. Hence one naturally comes up with the notion of goal reduction. In the introduction, we have already seen what the rules of goal reduction look like in the equational case. RED performs reduction steps by applying rewrite axioms (see the General Assumption). SUC, which determines when a goal reduction succeeds, admits not only reflexive equations as final goals, but arbitrary base theorems, thus exploiting previous knowledge about properties of the base specification. A variant of goal reduction allows to carry out reduction steps *modulo base theorems*. The *Congruence Class Approach* is not handled here, but studied in detail in Padawitz.²³

Weakening the notion of a succeeding goal reduction in the way described poses some problems, which can be managed by requiring that the terms in a reduced goal be *RAX-normal forms*. This excludes reflexive equations like $t \equiv t$ from the set of reduced goals whenever t can be rewritten by RAX. In summary, the two rules of goal reduction read as follows.

Definition. For all goals (or terms) δ , let $\text{single}(\delta)$ be the set of variables of δ , which occur exactly once in δ . Given a set R of conditional equations, the *goal reduction calculus w.r.t. R* consists of two derivation rules:

- (RED) for all goals δ , $x \in \text{single}(\delta)$, $u \equiv u' \Leftarrow \vartheta \in R$ and substitutions f ,
 $\delta[u[f]/x] \vdash \delta[u'[f]/x] \cup \vartheta[f]$.
- (SUC) For all unconditional and *R-normal* base theorems γ , $\gamma \vdash \emptyset$.

Let \vdash_R denote the corresponding inference relation. Derivations via \vdash_R are called *R-reductions*. A goal γ admits a *successful R-reduction* if $\gamma \vdash_R \emptyset$. ■

What is an *R-normal* goal? To make this precise we must refer to conditional *term* reductions in the sense of (1):

Definition. Given a set R of conditional equations, the set of *conditional reductions* $t \rightarrow_R t' \Leftarrow \gamma$ is inductively defined as follows:

- For all terms t , $x \in \text{single}(t)$, $u \equiv u' \Leftarrow \vartheta \in R$ and substitutions f , $t[u[f]/x] \rightarrow_R t[u'[f]/x] \Leftarrow \vartheta[f]$.
- If $t \rightarrow_R t' \Leftarrow \gamma$ and $t' \rightarrow_R t'' \Leftarrow \gamma'$, then $t \rightarrow_R t'' \Leftarrow \gamma \cup \gamma'$.

A term t is an *R-normal form* if there is no conditional reduction $t \rightarrow_R t' \Leftarrow \vartheta$. A goal γ is *R-normal* if all its subterms are *R-normal*. A substitution f is *R-normal* if all its values are *R-normal*. ■

Both conditional reductions and goal reductions are necessary for defining (unconditional) *term* reductions:

Definition. Given a set R of conditional equations, *term reductions* are all expressions of the form $t \rightarrow_R t'$ such that $t = t'$ or there is a conditional reduction $t \rightarrow_R t' \Leftarrow \gamma$ such that γ admits a successful *R-reduction*. In the second case, we also write $t \rightarrow_R^+ t'$. ■

In some cases, one needs \rightarrow_R and \rightarrow_R^+ as relations on goals. So we write $\gamma \rightarrow_R \gamma'$ if $t \rightarrow_R t'$ for some $t, t', \gamma = \delta[t/x]$ and $\gamma' = \delta[t'/x]$ for some δ .

We hope to have given enough motivation for distinguishing between goal reduction, conditional reduction and term reduction. What remains to be questioned is whether γ in SUC needs to be *R-normal*. In fact, there are several reasons for this requirement. Other approaches avoid it, but handle only restricted classes of specifications such as *hierarchical* (cf. Zhang & Remy)²⁹ *simplifying* (cf. Kaplan),²⁰ *reductive* (cf. Jouannaud & Waldmann)¹⁷ or *normal* ones (cf. Dershowitz *et al.*).⁶

The first reason is a practical one. If SUC could be applied even if γ is not *R-normal*, goal reduction would lead to a larger search space: there will be subgoals where both SUC and RED apply to, and it might be necessary to pursue both ways. On the other hand, reducing the number of checks for the applicability of SUC might be desirable because they involve deciding the base theory.

The second reason for sticking to normal forms can be put forward already for unconditional equations. A set R of equations is Church–Rosser (see the introduction) if \rightarrow_R is *confluent*. Unfortunately, this does not hold any more if R includes base axioms or if we allow arbitrary base theorems as reduced goals. When adding the requirement that reduced equations be *R-normal*, however, the result is retained, provided that R is *BAX-compatible* and *normalising* (cf. Padawitz,²³ Proposition 7.4.1).

Definition. Given a set R of conditional equations, two terms u and u' are *R-convergent modulo BAX* if $u \rightarrow_R v$, $u' \rightarrow_R v'$ for some base theorem $v \equiv v'$. R is *confluent modulo BAX* if for each two ‘branching’ reductions $t \rightarrow_R u$ and $t \rightarrow_R u'$, u and u' are *R-convergent modulo BAX*. R is *BAX-compatible* if for all reductions $t \rightarrow_R u$ and base theorems $t \equiv u'$, u and u' are *R-convergent modulo BAX*.

A term (or goal) is *R-normalisable* if $t \rightarrow_R u$ for some *R-normal* term u . R is *normalising* if all terms are *R-normalisable*. ■

In Padawitz,²³ the notion of *BAX-compatibility* refers to a kind of base *reductions*, which is stronger than the one given here. For the purpose of this paper it is sufficient to keep to base *theorems*.

Confluence and *BAX-compatibility* are indeed essential for the Church–Rosser property. The requirement that R be *normalising*, however, can be weakened by confining the Church–Rosser property to *normalisable* goals.

4. THE CHURCH–ROSSER PROPERTY

Definition. Given a set R of conditional equations, AX is *R-Church–Rosser* if all unconditional and *R-normalisable* AX -theorems admit a successful *R-reduction*. ■

By this definition, the Church–Rosser property of AX does no longer ensure that non-normalisable AX -theorems have a proof by reduction. But this seems to be a less serious restriction than presupposing that R is *normalising*.

The R-Church–Rosser property of AX implies that R-normal forms are unique ‘modulo BAX’:

Proposition 4.1. If AX is R-Church–Rosser, then each two AX-equivalent R-normal forms are BAX-equivalent. Moreover, the Church–Rosser property can be used for proving inductive theorems (cf. Section 2). For this purpose, R must be ‘functionally complete’ in the sense that all ground instances of the left-hand side of the theorem to be proved are reducible:

Definition. A conditional equation $e = t \equiv t' \leftarrow \gamma$ is *ground R-reducible* if for all ground substitutions f there is a term reduction $t[f] \rightarrow_R u$ whenever $\gamma[f]$ admits a successful $(R \cup \{e\})$ -reduction. ■

The following theorem (first proved in Jouannaud & Kounalis¹⁷ for the case of equations) allows us to reduce proofs by induction to proofs of the Church–Rosser property. Its assumptions are only needed for ground terms (goals): *ground normalising* means that only ground terms must have normal forms; *ground Church–Rosser* means that only ground theorems must admit successful reductions.

An inference relation \vdash is *well-founded* (Noetherian) if the rules of \vdash do not generate infinite derivation sequences.

Theorem 4.2. (Padawitz, Theorem 3.6).²⁵ Given a conditional equation e , suppose that $AX \cup \{e\}$ is ground $(RAX \cup \{e\})$ -Church–Rosser, \vdash_{RAX} is well-founded, $RAX \cup \{e\}$ is ground normalising and e is ground RAX-reducible. Then e is an inductive AX-theorem. ■

Let us now turn to a general criterion for the Church–Rosser property. In addition to confluence and BAX-compatibility we have to require that base theorems respect normal forms:

Definition. Given a set R of conditional equations, BAX *respects R-normal forms* if for all R-normal terms t , $BAX \vdash_c t \equiv u$ implies $u \rightarrow_R t'$ and $BAX \vdash_c t \equiv t'$ for some R-normal term t' . ■

In the case of equations, the general Church–Rosser criterion is the following.

Theorem 4.3 (Padawitz, Proposition 7.5.2).²³ Given that AX consists of unconditional equations, AX is RAX-Church–Rosser if RAX is confluent modulo BAX and BAX-compatible and BAX respects RAX-normal forms. ■

If R is normalising, then BAX-compatibility of RAX implies normal form respectation of BAX, and the converse of Theorem 4.3 holds true as well.

Before presenting the generalisation of Theorem 4.3 to *conditional* equations let us point out a third reason for calling goal reductions successful only if they end up with normal forms. Given an equation $t \equiv t' \in R$ and a term reduction $v \rightarrow_R v'$, there are two *non-overlapping* reductions

$$t[v/x] \rightarrow_R t'[v/x] \quad \text{and} \quad t[v/x] \rightarrow_R t[v'/x]. \quad (1)$$

The ‘reducts’ $t'[v/x]$ and $t[v'/x]$ are convergent: both terms can be reduced to $t'[v'/x]$. This simple fact allows us to conclude the confluence of R from the convergence of finitely many *critical* (term) *pairs*.

Unfortunately, the argument does not apply to *conditional* equations, unless we add the normal form requirement. If $t \equiv t'$ has a premise, say $u \equiv u'$, we obtain the *conditional* reduction

$$t[v/x] \rightarrow_R t'[v/x] \leftarrow (u \equiv u')[v/x] \quad (2)$$

and, if the premise instance $(u \equiv u')[v/x]$ admits a successful R-reduction, (1) holds true as in the equational case. Again, $t'[v/x]$ can be reduced to $t'[v'/x]$. However, the complementary reduction $t[v'/x] \rightarrow_R t'[v'/x]$ depends on a successful R-reduction of the new premise instance $(u \equiv u')[v'/x]$.

Since $(u \equiv u')[v/x]$ admits a successful R-reduction, there are term reductions $u[v/x] \rightarrow_R t_0$ and $u'[v/x] \rightarrow_R t_1$ and an R-normal base theorem $t_0 \equiv t_1$. Furthermore, $v \rightarrow_R v'$ implies $u[v/x] \rightarrow_R u[v'/x]$ and $u'[v/x] \rightarrow_R u'[v'/x]$. Hence, assuming that the premise terms of (2), $u[v/x]$ and $u'[v/x]$, are ‘smaller’ than the left-hand side instance of (2), $t[v/x]$, we conclude by the induction hypothesis that t_0 and $u[v'/x]$ as well as t_1 and $u'[v'/x]$ are R-convergent modulo BAX, i.e. there are reductions $t_0 \rightarrow_R t'_0$, $u[v'/x] \rightarrow_R t'_0$, $t_1 \rightarrow_R t'_1$ and $u'[v'/x] \rightarrow_R t'_1$ as well as base theorems $t'_0 \equiv t'_0$ and $t'_1 \equiv t'_1$.

Since t_0 and t_1 are R-normal, we actually have $t_0 = t'_0$. Provided that BAX respects R-normal forms, t'_0 and t'_1 are R-normal as well and thus yield a successful R-reduction of $(u \equiv u')[v'/x]$.

Without the normal form requirement this argument would not work and thus we could not ensure that non-overlapping reductions have convergent reducts.

In order to generalise Theorem 4.3 to conditional equations, replacing the conclusion instance of an axiom by the corresponding premise instance must preserve normalisability.

Definition. Let R be a set of conditional equations. AX *preserves R-normalisability* if for all substitutions f the following holds true:

- For all $q \leftarrow \vartheta \in AX$, if $q[f]$ is R-normalisable and $AX \vdash_c \vartheta[f]$, then $\vartheta[f]$ is R-normalisable too.
- For all $u \equiv u' \leftarrow \vartheta \in AX$, terms t and $x \in \text{var}(t)$, if $t[u/x]$ or $t[u'/x]$ is R-normalisable and $AX \vdash_c \vartheta[f]$, then $\vartheta[f]$ is R-normalisable as well. ■

Theorem 4.4. (Padawitz, Theorem 7.8.2).²³ Suppose that RAX is confluent modulo BAX and BAX-compatible, BAX respects RAX-normal forms, AX preserves RAX-normalisability and for all $u \equiv u' \leftarrow \vartheta \in RAX$, the leftmost symbol of u is not a base symbol. Then AX is RAX-Church–Rosser. ■

For proving inductive theorems (cf. Theorem 4.2) or guaranteeing the completeness of narrowing for ground substitutions (cf. Theorem 6.1) we only need the *ground* Church–Rosser property. For that purpose it is sufficient to assume the conditions of Theorem 4.4 only for ground terms (goals).

The practical applicability of Theorem 4.4 depends on decidable criteria for its assumptions. Sometimes the one or the other vanishes completely. For instance, if BAX is empty, then BAX-compatibility of RAX and normal form respectation of BAX hold true trivially. However, in this case we would have to show that the whole set of axioms is confluent (modulo the empty set).

If RAX is normalising, then, of course, AX preserves RAX-normalisability, and BAX-compatibility of RAX implies normal form respectation of BAX. Another criterion for the latter property consists of three conditions:

- For all $u \equiv u' \leftarrow \vartheta \in RAX$, the leftmost symbol of u is not a base symbol,
- for all $u \equiv u' \leftarrow \vartheta \in BAX$, $\text{var}(u) = \text{var}(u')$,
- all RAX-normal terms are base terms.

As mentioned above, the proof of confluence (and BAX-compatibility) must be reduced to the convergence of a finite number of *critical (term) pairs*. However, this criterion works only under additional assumptions like *left-linearity* of RAX or well-foundedness of \vdash_{RAX} (see above). Left-linearity, the property that a variable does not occur twice in the left-hand side of a rewrite axiom, can be avoided if one turns to the Congruence Class Approach (cf. Section 3). Well-foundedness of \vdash_{RAX} is not necessary if the critical pairs are convergent in a very strong sense, which nearly means that RAX is *non-overlapping* (cf. *strong confluence* in Padawitz).²³ If \vdash_{RAX} is well-founded, then RAX is normalising and thus AX preserves RAX-normalisability, which is another assumption of Theorem 4.4.

Sometimes these criteria can be fulfilled by changing the set of base axioms or by adding further axioms. *Knuth–Bendix completion* is the way of deriving such axioms from the set of non-convergent critical pairs. A special case is *inductive completion* where the Church–Rosser property is used for proving inductive theorems (cf. Theorem 4.2). Here the convergence checks for critical pairs often correspond to the cases of an explicit proof by induction (cf. Fribourg,⁸ Küchlin,²¹ Hofbauer and Kutsche¹⁴ and Padawitz²⁵).

In fact, many specifications do not pose serious problems concerning the choice and proof of suitable Church–Rosser criteria. For instance, if RAX is non-overlapping and left-linear and \vdash_{RAX} is well-founded, the proof obligation concentrates on the critical pair criteria for BAX-compatibility (cf. Padawitz,²³ Exercise 9.9.2). Nevertheless, sticking to a single set of conditions would exclude specifications whose axiom sets are Church–Rosser, but do not satisfy some of the conditions. In Padawitz,²³ Section 9.9, we have presented an algorithm that guides through the variety of Church–Rosser criteria.

5. REDUCTION STRATEGIES

Besides serving as a criterion for the Church–Rosser property, a confluent set of conditional equations allows us to restrict the applications of RED to a predefined strategy such as ‘innermost’, ‘outermost’, etc. Let us define a strategy as a function S from the set of goals to the set of positions. A *position of a goal* γ is a pair (δ, t) consisting of a goal δ with the distinguished variable x_0 and a term t such that $\delta[t/x_0] = \gamma$.

Definition. Given a conditional equation $e = u \equiv u' \Leftarrow \vartheta$ and a substitution f , the position $(\delta, u[f])$ is called a *reduction redex* of the goal $\delta[u[f]/x_0]$ *induced by* e . Given a set R of conditional equations, a position (δ, t) is an *R-reduction redex* if it is induced by some $e \in R$. If $(\delta, u[f])$ is a reduction redex induced by $u \equiv u' \Leftarrow \vartheta \in R$ and $\vartheta[f]$ admits a successful R-reduction, then $(\delta, u[f])$ is called *R-feasible*.

A function S from the set of goals to the set of positions is an *R-reduction strategy* if for all goals γ ,

- $S(\gamma)$ is a position of γ ,
- $S(\gamma)$ is an R-feasible reduction redex of γ whenever γ has an R-feasible reduction redex. ■

The restriction of RED to applications controlled by S reads as follows:

(RED-S) for all goals γ , $u \equiv u' \Leftarrow \vartheta \in R$ and

substitutions f such that for some δ, f , $S(\gamma) = (\delta, u[f])$, $\delta \vdash \delta[u[f]/x_0] \cup \vartheta[f]$.

Let $\vdash_{\text{R-S}}$ denote the inference relation generated by RED-S and SUC. A goal γ *admits a successful S-controlled R-reduction* if $\gamma \vdash_{\text{R-S}} \emptyset$.

Theorem 5.1. (Padawitz,²³ Theorem 7.9.2). Let R be a set of conditional equations and S be an R-reduction strategy such that R is confluent modulo BAX, BAX respects R-normal forms and $\vdash_{\text{R-S}}$ is well-founded. Then a goal γ admits a successful S-controlled R-reduction if γ admits any successful R-reduction. ■

If R is confluent modulo the empty set of axioms, the assumption ‘BAX respects R-normal forms’ can be dropped. If γ is a *ground* goal, it is sufficient to assume that R is confluent on *ground* terms and that the reduction strategy is defined only on *ground* goals.

The strategies defined here depend on redex positions, but neither attach priorities to the elements of R nor take into account the history of reduction sequences. They only consider the actual goal when determining the next reduction step. An important ‘computation rule’ which uses information about previous reduction steps is *basic reduction* (cf. Hullot,¹⁵ Section 4).¹⁵ The corresponding restriction of RED transforms only those reduction redices (δ, t) where t is not a subterm of the term substituted for some variable occurring in the right-hand side of the axiom which caused the preceding reduction step. Consequently, each goal γ in a reduction sequence must be associated with the location in γ of the respective right-hand side. Provided that \vdash_{R} is well-founded, basic reduction is complete with respect to goal reduction: A goal admits a successful basic R-reduction if it admits any successful R-reduction (cf. Padawitz, Lemma 7.10.2).²³

6. NARROWING

Goal reduction is a calculus for *proving* goals, narrowing adds the possibility of substituting for variables and thus extends goal reduction to a calculus for *solving* goals. The purpose of instantiating a goal γ is to complete a reduction redex only a prefix of which occurs in γ . The substitution must be *normal* (cf. Section 3) so that the prefix cannot be a empty. It must contain at least the leftmost function symbol of the left-hand side of an equation applicable to γ . Otherwise every reduction redex could be generated, just by replacing a variable of γ with the left-hand side of an arbitrary equation. Of course, the restriction to normal substitutions entails that we obtain only normal solutions of the initial goal. Under the assumption that all objects defined by the specification have normal form representations this is not really a restriction because then the set of normal solutions covers the set of all solutions.

Definition. A substitution f is an *AX-solution* of a goal γ if $\gamma[f]$ is an AX-theorem (cf. Section 2). ■

In consequence of the fact that narrowing extends goal reduction, the completeness of narrowing with respect to AX-solutions can be guaranteed only if AX is RAX–Church–Rosser.

A narrowing step transforms a pair $\langle \gamma, f \rangle$ consisting of a goal γ and a substitution f into a pair of the form $\langle \gamma', f[g] \rangle$ where $f[g]$ denotes the *instance of* f by g , i.e. for all variables x , $f[g](x)$ is defined as $f(x)[g]$ (cf. Section 1).

Instances give rise to the *subsumption relation*: A term (or substitution) t *subsumes* a term t' if $t' = t[g]$ for some g . Up to a renaming of variables, the subsumption relation is a partial order. Given a term t , the least elements in the set of unifiers of t (cf. Section 1) are called *most general unifiers* of t .

We assume that the variables of a goal subjected to a narrowing step belong to a set GV of variables, which do not occur in axioms. If a narrowing step brings axiom variables into a goal, they must be renamed before the derivation continues. g/GV denotes the functional restriction of g to GV . id is the identity on X .

Definition. Given a set R of conditional equations, the *narrowing calculus w.r.t. R* consists of two derivation rules:

- (NAR) For all goals δ , $x \in \text{single}(\delta)$, $t \in T(\text{SIG})-X$, $u \equiv u' \Leftarrow \vartheta \in R$, substitutions f and most general unifiers g of t and u ,
 $\langle \delta[t/x], f \rangle \vdash \langle \delta[u'/x][g] \cup \vartheta[g], f[g/GV] \rangle$.
- (SUC-N) For all unconditional R -normal BAX-theorems of the form $\gamma[g]$, $\langle \gamma, f \rangle \vdash \langle \emptyset, f[g] \rangle$.

Let \vdash_{R-N} denote the corresponding inference relation. Derivations via \vdash_{R-N} are called *R-narrowing expansions*. A substitution f is an *R-narrowing solution* of a goal γ if $\langle \gamma, id \rangle \vdash_{R-N} \langle \emptyset, f \rangle$. ■

Theorem 6.1. (Padawitz, Theorem 8.2.7).²³ Suppose that AX is RAX-Church–Rosser. Let γ be a goal and f be a RAX-normal substitution such that $\gamma[f]$ is RAX-normalisable. Then f is an AX-solution of γ iff γ has a RAX-narrowing solution, which subsumes f . ■

For ground substitutions, it is sufficient to assume that AX is ground RAX-Church–Rosser (cf. Section 4). The ground RAX-Church–Rosser property of AX implies that, given a ground term t , all RAX-normal forms, which are AX-equivalent to t , are BAX-equivalent (cf. Proposition 4.1). The set $NF(t)$ of these normal forms can be computed by narrowing: Theorem 6.1 implies that the elements of $NF(t)$ are subsumed by RAX-narrowing solutions of the equation $t \equiv x$ (with x being a variable), and, conversely, every RAX-narrowing solution of $t \equiv x$ belongs to $NF(t)$.

In Section 4, we mentioned the method of inductive completion for proving inductive theorems (cf. Section 2). A drawback of this method is the fact that the theorem to be proved, say e , must be considered as a rewrite axiom (cf. Theorem 4.2). If, on the other hand, narrowing is used for inductive proofs, the RAX-Church–Rosser property is yet crucial, but e is not submitted to Church–Rosser criteria. (For comparisons of inductive completion with other inductive proof methods, see Garland and Guttag¹⁰ and Padawitz^{24, 25}.)

Let us only consider the unconditional case, i.e., e is an atom. (The generalisation to *conditional* theorems is still under way; cf., e.g. Padawitz²⁴.) We may have several atoms which can only be proved simultaneously. So the question is whether a *goal*, say γ , is an inductive theorem. The following is an immediate consequence of Theorem 6.1:

Corollary 6.2. Suppose that AX is ground RAX-Church–Rosser and RAX is ground normalising. A goal γ is an inductive AX-theorem iff every ground RAX-normal substitution is subsumed by some RAX-narrowing solution of γ . ■

Well, but Corollary 6.2 does not provide an effective proof method. In many cases, one will not obtain a finite set of narrowing solutions that covers all ground normal substitutions. As a first improvement of the method, one may finish the derivation process as soon as a *ground complete* set of goal-substitution pairs has been accomplished.

Definition. A set GS of goal-substitution pairs is *ground complete* if for all ground substitutions f there are $\langle \delta, g \rangle \in GS$ and an AX-solution h of δ such that $g[h]$ is AX-equivalent to f . ■

Corollary 6.3. Suppose that AX is ground RAX-Church–Rosser and RAX is ground normalising. A goal γ is an inductive AX-theorem if and only if there is a set of RAX-narrowing expansions

$$\begin{aligned} \langle \gamma, id \rangle &\vdash_{RAX-N} \langle \delta_1, g_1 \rangle \\ \langle \gamma, id \rangle &\vdash_{RAX-N} \langle \delta_2, g_2 \rangle \\ &\dots \end{aligned}$$

such that $\{\langle \delta_1, g_1 \rangle, \langle \delta_2, g_2 \rangle, \dots\}$ is ground complete. ■

The next difficulty comes up when *lemmas* must be applied to prove an inductive theorem. This difficulty can be addressed differently. One may generalise Corollary 6.2 by taking into account that the inductive theory operator is idempotent, i.e. given a set L of inductive AX-theorems, a clause e is an inductive AX-theorem iff e is an inductive $(AX \cup L)$ -theorem (cf. Padawitz, Corollary 2.3).²⁴

Corollary 6.4. Suppose that L is a set of inductive AX-theorems, AX is ground $(RAX \cup L)$ -Church–Rosser and $RAX \cup L$ is ground normalising. A goal γ is an inductive AX-theorem iff every ground $(RAX \cup L)$ -normal substitution is subsumed by some $(RAX \cup L)$ -narrowing solution of γ . ■

However, this is a bad solution because it enforces the treatment of lemmas as rewrite axioms: the conditions on RAX become conditions on $RAX \cup L$. In fact, sticking to Corollary 6.4 goes into the direction of inductive completion. An alternative solution is to keep the lemmas away from the Church–Rosser property by using *resolution* and *paramodulation* for the application of lemmas and reserving narrowing for the application of axioms. Paramodulation generalises narrowing in that it allows applying conditional equations not only from left to right, but also from right to left:

- (PAR) For all goals δ , $x \in \text{single}(\delta)$, $t \in T(\bar{\text{SIG}})-X$, $u \equiv u' \Leftarrow \vartheta$ (or $u' \equiv u \Leftarrow \vartheta \in L$), substitutions f and most general unifiers g of t and u ,
 $\langle \delta[t/x], f \rangle \vdash \langle \delta[u'/x][g] \cup \vartheta[g], f[g/GV] \rangle$

Resolution is provided for applying lemmas which are not conditional equations;

- (RES) For all goals γ , atoms $p, q \Leftarrow \vartheta \in L$, substitutions f and most general unifiers g of p and q ,
 $\langle \gamma \cup \{p\}, f \rangle \vdash \langle \gamma[g] \cup \vartheta[g], f[g/GV] \rangle$.

Let \vdash_{R-L-N} denote the inference relation generated by NAR, SUC-N, PAR and RES. (R are axioms, L lemmas.) Derivations via \vdash_{R-L-N} are called *R-L narrowing expansions*.

Corollary 6.5. Suppose that L is a set of inductive AX-theorems, AX is ground RAX-Church–Rosser and RAX

is ground normalising. A goal γ is an inductive AX-theorem iff there is a set of RAX-L narrowing expansions

$$\langle \gamma, \text{id} \rangle \vdash_{\text{RAX-L-N}} \langle \delta_1, g_1 \rangle$$

$$\langle \gamma, \text{id} \rangle \vdash_{\text{RAX-L-N}} \langle \delta_2, g_2 \rangle$$

...

such that $\{\langle \delta_1, g_1 \rangle, \langle \delta_2, g_2 \rangle, \dots\}$ is ground complete. ■

Corollary 6.5 is not sufficient either. It is of practical use only if one gets by with finitely many expansions to obtain a ground complete set of goal-substitution pairs. If γ is a ‘proper’ inductive theorem, however, applying induction hypotheses is necessary to get a finite proof. Induction hypotheses of γ can be treated like lemmas, i.e. by resolution and paramodulation, but here we resolve or paramodulate on γ itself! When applying γ a subgoal of the form $\{t \gg t'\}$ is generated, expressing that the actual instance of γ (representing the induction hypothesis) is ‘smaller’ than the instance of γ we are going to prove. So the specification must include a predicate \gg , defined as a well-founded relation on those ground terms which can be substituted for variables of γ . Assumed that these variables are given by the sequence z , z' is a copy of z and $\gamma' = \gamma[z'/x]$, the rules for applying induction hypotheses are as follows.

(PAR-IN) For all goals δ , $x \in \text{single}(\delta)$, $t \in T(\text{SIG})\text{-X}$, $u \equiv u'$ (or $u' \equiv u$) $\in \gamma'$, substitutions f and most general unifiers g of t and u , $\langle \delta[t/x], f \rangle \vdash \langle \delta[u'/x][g] \cup \{fz \gg z'\}, f[g|GV] \rangle$.

(RES-IN) For all goals γ , atoms $p, q \in \gamma'$, substitutions f and most general unifiers g of p and q , $\langle \gamma \cup \{p\}, f \rangle \vdash \langle \gamma[g] \cup \{fz \gg z'\}, f[g|GV] \rangle$.

Let $\vdash_{\text{R-L-}\gamma\text{-N}}$ denote the inference relation generated by NAR, SUC-N, PAR, RES, PAR-IN and RES-IN. (R are axioms, L lemmas, γ induction hypotheses.) Derivations via $\vdash_{\text{R-L-}\gamma\text{-N}}$ are called *R-L- γ -narrowing expansions*.

Only this calculus includes enough rules to offer reasonable chances for the termination of inductive proofs based on narrowing.

Corollary 6.6. Suppose that L is a set of inductive AX-theorems, AX is ground RAX-Church–Rosser and RAX is ground normalising. A goal γ is an inductive AX-theorem iff there is a set of RAX-L- γ -narrowing expansions

$$\langle \gamma, \text{id} \rangle \vdash_{\text{RAX-L-}\gamma\text{-N}} \langle \delta_1, g_1 \rangle$$

$$\langle \gamma, \text{id} \rangle \vdash_{\text{RAX-L-}\gamma\text{-N}} \langle \delta_2, g_2 \rangle$$

such that $\{\langle \delta_1, g_1 \rangle, \langle \delta_2, g_2 \rangle, \dots\}$ is ground complete. ■

The proofs of Corollaries 6.5 and 6.6 have not yet been worked in detail. But we conjecture that they go analogously to the proof of Padawitz,²⁴ Theorem 4.7, which states a similar result for resolution and paramodulation (upon axioms) instead of narrowing.

7. NARROWING STRATEGIES

Extending a reduction strategy S (cf. Section 6) to a narrowing strategy requires that the redex selection of S be uniform in the sense that, given a goal γ , the selected redex location is the same for all instances of γ by R-normal substitutions.

Definition. Given a conditional equation $e = u \equiv u' \Leftarrow \vartheta$, $t \in T(\text{SIG})\text{-X}$ and a unifier of t and u , the position (δ, t)

is called a *narrowing redex* of the goal $\delta[t/x_0]$ induced by e . Given a set R of conditional equations, a position (δ, t) is an *R-narrowing redex* if it is induced by some $e \in R$.

A function S from the set of goals to the set of positions is an *R-narrowing strategy* if for all goals γ ,

- S(γ) is a position of γ ,
- S(γ) is an R-narrowing redex of γ whenever γ has an R-narrowing redex,
- redex selection is uniform, i.e. for all R-normal substitutions f , $S(\gamma[f]) = S(\gamma)[f]$, whenever S(γ) is an R-narrowing redex. ■

The restriction of NAR to applications controlled by S reads as follows:

(NAR-S) For all goals γ , $u \equiv u' \Leftarrow \vartheta \in R$ and substitutions f such that for some δ, t , $S(\gamma) = (\delta, t)$ and f is a unifier of t and u , $\langle \gamma, t \rangle \vdash \langle \delta[u'/x_0][g] \cup \vartheta[g|GV] \rangle$.

Let $\vdash_{\text{R-S-N}}$ denote the inference relation by NAR-S and SUC-N. A substitution f is an *S-controlled R-narrowing solution* of a goal γ if $\langle \gamma, \text{id} \rangle \vdash_{\text{R-S-N}} \langle \emptyset, f \rangle$. ■

When combining Theorems 5.1 and 6.1 one obtains

Theorem 7.1. (Padawitz, Theorem 8.3.5).²³ Suppose that AX is RAX-Church–Rosser, RAX is confluent modulo BAX and BAX respects RAX-normal forms. Let S be a RAX-reduction and -narrowing strategy such that $\vdash_{\text{RAX-S}}$ (cf. Section 5) is well-founded. Let γ be a goal and f be a RAX-normal substitution such that $\gamma[f]$ is RAX-normalisable. Then f is an AX-solution of γ iff γ has an S-controlled RAX-narrowing solution, which subsumes f . ■

Again, for ground substitutions f , all assumptions are only needed for ground terms (goals).

Note that the well-foundedness of $\vdash_{\text{R-S}}$ does not imply that R is normalising. But if R is normalising, the assumptions ‘R is confluent modulo BAX’ and ‘BAX respects R-normal forms’ can be dropped because they follow from the Church–Rosser property of AX.

The uniformity of redex selection is crucial for the completeness of a narrowing strategy. A local criterion for this property is the following: For all atoms p , goals γ , substitutions f and ground R-normal substitutions g such that $S(p) = (q, t)$ is an R-narrowing redex of p ,

- $S(\gamma \cup \{p\}) = (\gamma \cup \{q\}, t)$,
- $S(p[f]) = (q', t')$ implies that x_0 occurs ‘at the same place’ in q and q' ,
- there is an instance of the form $t[g] \equiv u \Leftarrow \vartheta$ of some $e \in R$.

The third condition entails a certain *functional completeness* of R: If (q, t) is an R-narrowing redex, i.e., if for some g there is an instance $t[g] \equiv u \Leftarrow \vartheta$ of some $e \in R$, then such an instance is required for *every* (ground R-normal) substitution g . Syntactical criteria can be found in Echahed,⁷ Section 3 and Padawitz,²³ Section 8.4.

Basic narrowing generalizes basic reduction (cf. Section 5) to narrowing. It confines narrowing steps to those redices (δ, t) where t is not a subterm of the term substituted for some variable occurring in the right-hand side of the axiom which caused the preceding narrowing step. Consequently, each goal-substitution pair in a narrowing sequence must be associated with the location in γ of the respective right-hand side. Provided that \vdash_{R} is well-founded, basic narrowing is complete with respect

to narrowing: Each R-normal R-narrowing solution of a goal γ is subsumed by a basic R-narrowing solution of γ , i.e. a solution obtained by a basic R-narrowing expansion (cf. Padawitz,²³ Theorem 8.6.5).

A further specialisation of narrowing comes up when narrowing steps are followed by term reductions according to a predefined *R-reduction mapping* $RM: T(SIG) \rightarrow T(SIG)$. RM may assign to a term its R-normal form if it exists, or an AX-equivalent base term. The only requirement to RM is that it maps each term t to a reduct of t , i.e. $t \rightarrow_R RM(t)$.

A narrowing step and the application of RM to the resulting goal is combined into a single rule:

(NAR-RM) For all goals δ , $x \in \text{single}(\delta)$, $t \in T(SIG)-X$, $u \equiv u' \Leftarrow \vartheta \in R$, substitutions f and most general unifiers g of t and u , $\langle \delta[t/x], f \rangle \vdash \langle RM(\delta[u'/x][g] \cup \vartheta[g]), f[gGV] \rangle$.

Let \vdash_{R-RM-N} denote the inference relation generated by NAR-RM and SUC-N. A substitution f is an *RM-reduced R-narrowing solution* of a goal γ if $\langle \gamma, id \rangle \vdash_{R-RM-N} \langle \emptyset, f \rangle$.

Theorem 7.2. (Padawitz,²³ theorem 8.7.3). given a set R of conditional equations such that R is confluent modulo BAX, BAX respects R-normal forms and \vdash_R is well-founded. Let RM be an R-reduction mapping. An R-normal substitution f is an R-narrowing solution of a goal γ iff f is an RM-reduced R-narrowing solution of γ . ■

If R is confluent modulo the empty set of axioms, the assumption 'BAX respects R-normal forms' can be dropped.

S-controlled, basic and RM-reduced narrowing are in fact specialisations of the narrowing calculus. A rather different approach is pursued by *optimised narrowing*, which transforms a goal-substitution pair according to an *optimising function* before subjecting it to a narrowing step. Optimising functions need not be combinations of narrowing steps. Moreover, these functions may take into account not only the actual goal-substitution pair, but also its predecessors and sometimes even other narrowing expansions with the same initial goal-substitution pair. An optimising function may also cut off expansions which will fail eventually. So, given that GS denotes the set of goal-substitution pairs, an optimising function is a family $Op = \{Op(M) | M \subseteq GS\}$ of partial functions on GS . The corresponding narrowing expansions are best imagined as paths of an *optimised narrowing tree* that is given by a partial function $OpT: \mathbb{N}^* \rightarrow GS$, satisfying the following properties:

- $OpT(\varepsilon) = \langle \gamma, id \rangle$ for some γ . (At the root we start with the identity substitution.)
- For all $i, n \in \mathbb{N}$, $w \in \mathbb{N}^{\leq n}$, $OpT(wi)$ is undefined or there is an application of NAR leading from $OpT(w)$ to some $\langle \delta, f \rangle$ such that $OpT(wi) = Op(M)(\delta, f)$ for some $M \subseteq OpT(\mathbb{N}^{\leq n})$. (The optimisation of a goal-substitution pair at level $n+1$ takes into account only those goal-substitution pairs that were obtained up to level n .)

The correctness and completeness of optimised narrowing (with respect to AX-solutions) depends on local properties of the optimising function, which ensure that neither pure narrowing solutions are lost nor invalid

solutions are generated by inserting optimisation steps. (See Padawitz,²³ Section 8.9, for the details.)

The optimising functions which can be found in narrowing implementations like Hußmann,¹⁶ Rety *et al.*²⁷ and Fribourg⁹ are built up from elementary transformations like *goal subsumption*, *subsumption of solutions*, *expansion of variables*, *construction of substitutions*, *splitting*, *absorption and clash of equations* and *the rejection of non-narrowable goals*. Under certain conditions on the specification these optimising functions fulfil the local correctness conditions and, provided that AX is RAX-Church-Rosser, each RAX-normal AX-solution occurs in a corresponding optimised narrowing tree (cf. Padawitz, Theorem 8.9.3 and Section 8.10).²³

8. LAZY NARROWING

The elementary optimisations 'variable expansion' and 'equation splitting' are in fact rules for *lazy resolution* (cf. Padawitz, Section 5.5).²³ The attribute 'lazy' indicates that an occurrence in the actual goal of the leftmost symbol of an axiom is already sufficient for applying this axiom to the goal. A lazy inference rule *demand*s the full redex by creating an appropriate subgoal. The idea stems from Reddy.²⁶ The lazy narrowing calculus given below combines *lazy paramodulation* as it appears in Gallier & Snyder¹² and Hölldobler¹³ with the *outer narrowing* of You.²⁸ The rules are tailored to equational goals where the right-hand sides are base terms. Moreover, base equations are not allowed and R is supposed to be *innermost* and *ground term reducing*.

Definition. A set R of conditional equations is *ground term reducing* if all ground R-normal terms are base terms. A term t is *innermost* if its leftmost symbol is not a base symbol and all other symbols of t are base symbols. R is *innermost* if for all $t \equiv t' \Leftarrow \vartheta \in R$, t is innermost, $\text{var}(t') \cup \text{var}(\vartheta) \subseteq \text{var}(t)$ and for all $u \equiv u' \in \vartheta$, u' is a base term. ■

Definition. Given a set R of conditional equations, the *lazy narrowing calculus w.r.t. R* consists of SUC-N (cf. Section 6.1) and three further derivation rules:

- (L-NAR) For all goals γ , functions F , equations of the form $Ft \equiv u$ and $Fu' \equiv t' \Leftarrow \vartheta \in R$,
 $\langle \gamma \cup \{Ft \equiv u\}, f \rangle \vdash \langle \gamma \cup \{t \equiv u', t' \equiv u\} \cup \vartheta, f \rangle$.
- (SPLIT) For all goals γ , functions F and equations of the form $Ft \equiv Fu$,
 $\langle \gamma \cup \{Ft \equiv Fu\}, f \rangle \vdash \langle \gamma \cup \{t \equiv u\}, f \rangle$.
- (BASE) For all goals γ , base functions F , $x \in X$ and equations of the form $Ft \equiv x$,
 $\langle \gamma \cup \{Ft \equiv x\}, f \rangle \vdash \langle \gamma \cup \{t \equiv z\} [Fz/x], f[Fz/x] \rangle$
 where z is a sequence of variables not occurring in $\gamma \cup \{Ft \equiv x\}$.

Let \vdash_{R-LN} denote the corresponding inference relation. Derivations via \vdash_{R-LN} are called *lazy R-narrowing expansions*. A substitution f is a *lazy R-narrowing solution* of a goal γ if $\langle \gamma, id \rangle \vdash_{R-LN} \langle \emptyset, f \rangle$. ■

A more general lazy narrowing rule was proposed by Martelli *et al.*²² where F need not be the leftmost symbol of the actual goal:

- (MMR) For all goals γ , functions F , equations of the

$$\begin{aligned} &\text{form } v[Ft/x] \equiv u \text{ and } Fu' \equiv t' \Leftarrow g \in R, \\ &\langle \gamma \cup \{v[Ft/x] \equiv u\}, f \rangle \\ &\quad \vdash \langle \gamma \cup \{t \equiv u', v[t'/x] \equiv u\} \cup g, f \rangle. \end{aligned}$$

L-NAR only admits the case $v = x$. By admitting proper subterms of the goal as redices,²² is somewhat closer to NAR than L-NAR. However,²² may lead to more (eventually failing) expansions than L-NAR because it is applicable to each goal that anywhere contains the leftmost symbol of an axiom.

The following completeness result for lazy narrowing is a special case of Padawitz, Theorem 8.11.6.²³

Theorem 8.1. Suppose that AX is RAX-Church-Rosser, BAX does not contain conditional equations, RAX is innermost, ground term reducing and confluent on ground terms (cf. Section 3) and \vdash_{RAX} is

well-founded. Let γ be a goal consisting of equations with base terms as right-hand sides. A ground RAX-normal substitution f is an AX-solution of γ iff γ has a lazy RAX-narrowing solution, which agrees with f on the variables of γ . ■

The rules of the lazy narrowing calculus preserve the structure of goals presumed by Theorem 8.1. Since RAX is innermost, we can be sure that L-NAR is only applied to a *non-base* function F , whereas SPLIT will only encounter a *base* function F . On the other hand, there is no rule for transforming an equation of the form $Ft \equiv Gu$ with different base symbols F and G . At least here the efficiency of lazy narrowing improves over narrowing because such an equation indicates that the expansion where it occurs will fail and thus need not be continued (cf. Dincbas & van Hentenryck,⁵ Section 5).

REFERENCES

1. J. A. Bergstra and J. W. Klop, Conditional rewrite rules: confluence and termination, *J. Comp. and Syst. Sciences* **32**, 323–362 (1986).
2. G. Birkhoff, On the structure of abstract algebras, *Proc. Cambridge Philos. Society* **31**, 433–454 (1935).
3. R. S. Boyer and J. S. Moore, *A Computational Logic*, Academic Press (1979).
4. R. M. Burstall, Proving properties of programs by structural induction, *Comp. J.* **12**, 41–48 (1969).
5. M. Dincbas and P. van Hentenryck, Extended unification for the integration of functional programming into logic programming, *J. Logic Programming* **4**, 199–227 (1987).
6. N. Dershowitz, M. Okada and G. Sivakumar, Confluence of conditional rewrite systems, *Proc. Conditional Term Rewriting Systems '87*, Springer Lecture Notes in Computer Science **308**, 31–44 (1988).
7. R. Echahed, On completeness of narrowing strategies, *Proc. CAAP '88*, Springer Lecture Notes in Computer Science **299**, 89–101 (1988).
8. L. Fribourg, A strong restriction of the inductive completion procedure, *Proc. ICALP '86*, Springer Lecture Notes in Computer Science **226**, 105–115 (1986).
9. L. Fribourg, *Prolog with Simplification*, ESPRIT Report METEOR, L.I.T.P., Paris (1986).
10. S. J. Garland and J. V. Guttag, Inductive methods for reasoning about *Abstract Data Types*, *Proc. POPL '88*, 219–228 (1988).
11. J. A. Goguen and J. Meseguer, Completeness of many-sorted equational logic, *Houston Journal of Mathematics* **11**, 307–334 (1985).
12. J. H. Gallier and W. Snyder, A general complete E-unification procedure, *Proc. RTA '87*, Springer Lecture Notes in Computer Science **256**, 216–227 (1987).
13. S. Hölldobler, A unification algorithm for confluent theories, *Proc. ICALP '87*, Springer Lecture Notes in Computer Science **267**, 31–41 (1987).
14. D. Hofbauer and R. Kutsche, Proving inductive theorems based on term rewriting systems, *Proc. Algebraic and Logic Programming, Math. Research* **49**, Akademie-Verlag Berlin, 180–190 (1988).
15. J. M. Hullot, Canonical Forms and Unification, *Proc. CADE '80*, Springer Lecture Notes in Computer Science **87**, 318–334 (1980).
16. H. Hußmann, Unification in conditional-equational theories, *Proc. EUROCAL '85*, Springer Lecture Notes in Computer Science **204**, 543–553 (1985).
17. J.-P. Jouannaud and E. Kounalis, Automatic proofs by induction in equational theories without constructors, *Proc. LICS '86*, 358–366 (1986).
18. J.-P. Jouannaud and B. Waldmann, Reductive conditional term rewriting systems, *Proc. Conf. Formal Description of Programming Concepts III*, North-Holland, 223–244 (1986).
19. S. Kaplan, Conditional Rewrite Rules, *Theoretical Computer Science* **33**, 175–194 (1984).
20. S. Kaplan, Simplifying conditional term rewriting systems: unification, termination and confluence, *J. Symbolic Computation* **4**, 295–334 (1987).
21. W. Kuchlin, Inductive completion by ground proof transformation, *Proc. Resolution of Equations in Algebraic Structures*, Austin (1987).
22. A. Martelli, C. Moiso and G. F. Rossi, An algorithm for unification in equational theories, *Proc. IEEE Logic Programming '86*, 180–186 (1986).
23. P. Padawitz, *Computing in Horn Clause Theories*, *EATCS Monographs on Theor. Comp. Sci.* **16**, Springer (1988).
24. P. Padawitz, Inductive proofs by resolution and paramodulation, *Proc. CAAP '89* (1988).
25. P. Padawitz, *Proof by Consistency of Conditional Equations*, unpublished manuscript (1988).
26. U. Reddy, Narrowing as the operational semantics of functional languages, *Proc. IEEE Logic Programming '85*, 138–151 (1985).
27. P. Rety, C. Kirchner, H. Kirchner and P. Lescanne, NARROWER: A new algorithm for unification and application to logic programming, *Proc. RTA '85*, Springer Lecture Notes in Computer Science **202**, 141–157 (1985).
28. J. H. You, Outer narrowing for equational theories based on constructors, *Proc. ICALP '88*, Springer Lecture Notes in Computer Science **317**, 727–741 (1988).
29. H. Zhang and J.-L. Remy, Contextual rewriting, *Proc. RTA '85*, Springer Lecture Notes in Computer Science **202**, 46–62 (1985).