A Constructor-Based Approach for Positive/Negative-Conditional Equational Specifications

Claus-Peter Wirth, Bernhard Gramlich

Fachbereich Informatik, Universität Kaiserslautern
Postfach 3049, W-6750 Kaiserslautern
Germany

{wirth, gramlich}@informatik.uni-kl.de

May 4, 1992

Abstract: We study algebraic specifications given by finite sets R of positive/negative-conditional equations (i. e. universally quantified first-order implications with a single equation in the succedent and a conjunction of positive and negative (i. e. negated) equations in the antecedent). The class of models of such a specification R does not contain in general a minimum model in the sense that it can be mapped to any other model by some homomorphism. We present a constructor-based approach for assigning appropriate semantics to such specifications. We introduce two syntactic restrictions: firstly, for a condition to be fulfilled we require the evaluation values of the terms of the negative equations to be in the set of evaluation values of constructor ground terms; secondly, we restrict the constructor rules to have "Horn"-form and to be "constructor-preserving". A reduction relation for R is defined, which allows to generalize the fundamental results for positive-conditional rewrite systems, which does not need to be noetherian or restricted to ground terms, and which is monotonic w. r. t. the extension of the specification. Under the assumption of confluence, the factor algebra of the ground term algebra modulo the congruence of our reduction relation is a minimal model which is (beyond that) the minimum of all models that do not identify more constructor ground terms than necessary. To achieve decidability of reducibility we define several kinds of compatibility of R with a reduction ordering and present a complete critical-pair test à la Knuth-Bendix for the confluence of our reduction relation.

This research was supported by the Deutsche Forschungsgemeinschaft, SFB 314 (D4-Projekt)

Contents

1 Introduction

We study algebraic specifications with finite sets of positive/negative-conditional equations. We present a new uniform (i. e. not using "built-in"s or specification hierarchies) constructor-based approach for assigning appropriate semantics to such specifications. A reduction relation is defined which allows us to generalize the fundamental results for positive-conditional rewrite systems (cf. [22, 12]). Under the assumption of confluence, the factor algebra of the ground term algebra modulo the congruence of our reduction relation is an (up to isomorphism) uniquely determined minimal model, which (as developed in Wirth[33]) can be used to establish inductive validity and allows a straightforward abstraction for the treatment of non-ground equational clauses by contextual rewriting.

In spite of the obvious importance of positive/negative-conditional equations, their use for algebraic specification has rarely been studied so far compared to the use of merely positive conditional equations. The reason for this might be that they must be at least as complicated as merely positive conditional equations and have the serious problem of not exhibiting a straightforward algebraic behaviour because they do not have an initial algebra semantics as given for merely positive conditional equations: The class of models of a finite set R of positive/negative-conditional equations does not contain in general a minimum model in the sense that it can be mapped to any other model by some homomorphism; in other words, the set of congruences on the ground term algebra which fulfill the equations of the specification does not contain in general a minimum w. r. t. set-inclusion. The most promising attempt in literature to overcome this problem has been that in Kaplan[23]. There, one of the minimal ("quasi-initial") models (or congruence relations) is distinguished from the other minimal models by the use of control information extracted from the rules, which must be compatible with a noetherian ordering. In addition, Kaplan gives a straightforward ground term reduction relation. However, the approach of Kaplan[23] violates the paradigm of separation of logic and control and does not allow to express the distinction of the minimal model without the control part of the specification. Therefore, one loses all the nice aspects of the pure logic view on the specification which becomes a mere program instead (cf. sect. 3).

For that reason, we choose a new different approach. Instead of using control information we introduce two syntactic restrictions: Firstly, for a condition to be fulfilled we require the evaluation values of the terms of the negative equations to be in the set of evaluation values of constructor ground terms by adding condition literals expressing this property. Secondly, we restrict the constructor rules (which express equalities among the constructor terms) to have "Horn"-form and to be "constructor-preserving". We can then carry over the results of Kaplan[23] without using control information or noetherian orderings anymore (cf. sect. 5).

As a consequence, our reduction relation does not need to be noetherian or to be restricted to ground terms. Contrary to [23], we can show the monotony of this reduction relation w. r. t. the consistent extension of the specification. As in [23], under the assumption of confluence of our reduction relation, the factor algebra of the ground term algebra modulo the congruence of our reduction relation is a minimal model for our specification. Unlike [23], however, it is also the minimum of all models that do not identify more constructor ground terms than necessary.

To achieve decidability of reducibility we define several kinds of compatibility of R with a reduction ordering and present a complete critical-pair test à la Knuth-Bendix for the confluence of our reduction relation.

The following sections are organized as follows: Section 3 gives a motivating study of the semantic problems of positive/negative-conditional equations. In section 5 we define our reduction relation suitable for our syntax and semantics given in section 4. In section 7 we study orderings appropriate for guaranteeing termination of the evaluation of conditional rules and, moreover, study confluence and decidability of reducibility of rule systems that are compatible with such orderings. The proofs of our lemmas and theorems can be found in an appendix (sect. 10).

2 Basic Notions and Notations

In this section we describe our non-standard basic notions and notations (for sets, terms, orderings, reduction relations, and algebras) which are used throughout the following sections.

2.1 Sets and Classes

We use the following non-standard notation for classes A, B:

$$\begin{array}{rcll}
\mathbb{N}_+ & := & \{\ n \in \mathbb{N} \mid n \neq 0\ \} & \\
\mathrm{dom}(A) & := & \{\ a \mid \exists b : (a,b) \in A\ \} & (\textit{domain}) \\
\mathrm{ran}(A) & := & \{\ b \mid \exists a : (a,b) \in A\ \} & (\textit{range}) \\
\mathrm{field}(A) & := & \mathrm{dom}(A) \cup \mathrm{ran}(A) & \\
A(B) & := & \bigcap_{(B,C) \in A} C & (\textit{function application of } A)^1 \\
A[B] & := & \{\ b \mid \exists a \in B : (a,b) \in A\ \} & (\textit{relation application of } A) \\
\mathcal{F}(A) & := & \{\ S \mid\ S \subseteq A\ \wedge\ (S \text{ is finite})\ \ \} & \\
A^* & := & \{\ w : \{0,1,\ldots,n-1\} \to A \mid n \in \mathbb{N}\ \} & (\textit{set of words}) \\
A^+ & := & \{\ w : \{0,1,\ldots,n-1\} \to A \mid n \in \mathbb{N}_+\ \} & (\textit{set of nonempty words}) \\
\mathrm{id} & := & \{\ (a,a) \mid a = a\ \} & (\textit{identity-function}) \\
\mathcal{U} & := & \{\ a \mid a = a\ \} & (\textit{class of all sets}) \\
A^{-1} & := & \{\ (b,a) \mid (a,b) \in A\ \} & (\textit{reverse of a relation}) \\
A \circ B & := & \{\ (a,c) \mid \exists b :\ \ ((a,b) \in A\ \wedge\ (b,c) \in B)\ \} & (\textit{product of relations})^2 \\
A^{\circ(0)} & := & \mathrm{id} & \\
A^{\circ(i+1)} & := & A^{\circ(i)} \circ A\ \ \text{ for } i \in \mathbb{N} & (\textit{power of a relation}) \\
A^{\circledast} & := & \bigcup_{i \in \mathbb{N}} A^{\circ(i)} & \left(\begin{array}{c}\mathcal{U}\textit{-reflexive, transitive} \\ \textit{closure of a relation}\end{array}\right) \\
A^{\oplus} & := & \bigcup_{i \in \mathbb{N}_+} A^{\circ(i)} & \left(\begin{array}{c}\textit{transitive} \\ \textit{closure of a relation}\end{array}\right)
\end{array}$$

We use $\emptyset$ both for the empty set and the empty word. For application of the function f to i we write f_i besides $f(i)$, so that w_i denotes the[3] i^{th} "letter" of a word w. Speaking of a *relation* we always think of a binary relation unless indicated otherwise. We use $\uplus$ for the union of disjoint sets.

---

[1] As this set-theoretic definition might be difficult to understand, here is an equivalent one:

$$A(B) = \begin{cases} C & \text{if } \ B \in \mathrm{dom}(A)\ \wedge\ \forall D((B,D) \in A\ \Rightarrow\ D = C) \\ \mathcal{U} & \text{if } \ B \notin \mathrm{dom}(A) \\ \textit{“rubbish”} & \text{otherwise} \end{cases}$$

[2] This is very convenient for relations because $a\,A\,b\,B\,c$ implies $a\,(A \circ B)\,c$. It is, however, not generally considered the most convenient definition when A and B are used as functions because $B(A(a)) = b$ (and not $A(B(a)) = b$) implies $(A \circ B)(a) = b$, provided that $a \in \mathrm{dom}(A)$ and $A(a) \in \mathrm{dom}(B)$.

[3] Notice that we start with 0 instead of 1

2.2 Terms

We will consider terms of fixed arity over many-sorted signatures. A *signature*

$$\mathrm{sig} = (\mathrm{F}, \mathrm{S}, \alpha)$$

consists of an enumerable set of function symbols F, a finite set of sorts S (disjoint from F), and a computable arity-function $\alpha : \mathrm{F} \to \mathrm{S}^{+}$. For $f \in \mathrm{F}$: $\alpha(f)$ is the list of argument sorts augmented by the sort of the result of f; to ease reading we will add a ‘ $\longrightarrow$ ’ after a nonempty list of argument sorts. F has a decidable subset C of *constructor symbols*, which constitute their own signature $\mathrm{cons} = (\mathrm{C}, \mathrm{S}, \alpha|_{\mathrm{C}})$. The complement $\mathrm{D} = \mathrm{F} \setminus \mathrm{C}$ is called the set of *defined symbols*.

Example 2.1 (Signature with Constructor Subsignature)

$$\begin{array}{rcll}
\mathrm{C} & = & \{\mathsf{0}, \mathsf{s}, \mathsf{false}, \mathsf{true}, \mathsf{nil}, \mathsf{cons}\} & \\
\mathrm{D} & = & \{\mathsf{-}, \mathsf{memberp}, \mathsf{subsetp}\} & \\
\mathrm{S} & = & \{\mathsf{nat}, \mathsf{bool}, \mathsf{list}\} & \\
\alpha(\mathsf{0}) & = & & \mathsf{nat} \\
\alpha(\mathsf{s}) & = & \mathsf{nat} \quad \longrightarrow & \mathsf{nat} \\
\alpha(\mathsf{false}) & = & & \mathsf{bool} \\
\alpha(\mathsf{true}) & = & & \mathsf{bool} \\
\alpha(\mathsf{nil}) & = & & \mathsf{list} \\
\alpha(\mathsf{cons}) & = & \mathsf{nat\ list} \quad \longrightarrow & \mathsf{list} \\
\alpha(\mathsf{-}) & = & \mathsf{nat\ nat} \quad \longrightarrow & \mathsf{nat} \\
\alpha(\mathsf{memberp}) & = & \mathsf{nat\ list} \quad \longrightarrow & \mathsf{bool} \\
\alpha(\mathsf{subsetp}) & = & \mathsf{list\ list} \quad \longrightarrow & \mathsf{bool}
\end{array}$$

A variable-system for a signature sig is an S-sorted family of decidable sets which are mutually disjoint and disjoint from F. By abuse of notation we will use the symbol ‘X’ for an S-sorted family not only to denote the family $X = (X_s)_{s \in \mathrm{S}}$ itself, but also the union of its *ranges*: $\bigcup_{s \in \mathrm{S}} X_s$. As the basis for our terms throughout the whole paper we assume a fixed variable-system V with infinite ranges only. For denoting the family of variables occurring in a structure A (which might be a term or a list of terms, e. g.) we will use $\mathcal{V}(A)$. $\mathcal{T}(\mathrm{sig})$ denotes the S-sorted family of all (well-sorted) *terms* over sig and V, while $\mathcal{GT}(\mathrm{sig})$ denotes the S-sorted family of all *ground terms* over sig. Similarly, $\mathcal{T}(\mathrm{cons})$ denotes the S-sorted family of all *constructor terms*, while $\mathcal{GT}(\mathrm{cons})$ denotes the S-sorted family of all *constructor ground terms* which we always assume to have nonempty ranges only. We will denote all terms in prefix-form without any confusing notational sugar (such as parentheses or commas).

An *occurrence* (or *position*) is an element of $\mathbb{N}_{+}^{*}$. We write $p|q$ to express that neither p is a prefix of q, nor q a prefix of p. For a term $t \in \mathcal{T}(\mathrm{sig})$ we define the *set of its occurrences* (or *positions*) $\mathcal{O}(t)$ recursively by:

$$\begin{array}{rll}
\mathcal{O}(x) = \{\emptyset\} & & \text{for } x \in \mathrm{V} \\
\mathcal{O}(f t_1 \ldots t_n) = \{\emptyset\} \cup \bigcup\limits_{1 \leq i \leq n} \{\ iq \mid q \in \mathcal{O}(t_i)\ \} & & \text{for } f \in \mathrm{F};\ t_1, \ldots, t_n \in \mathcal{T}(\mathrm{sig})
\end{array}$$

For a term t we denote by t/p the subterm of t at occurrence p and by $t[\,p \leftarrow t'\,]$ the result of replacing t/p with t' at occurrence p in t. For $P \subseteq \mathcal{O}(t)$, $\forall p, q \in P : (p \neq q \;\Rightarrow\; p|q)$, we denote by $t[\,p \leftarrow t'_p \mid p \in P]$ the result of replacing for each $p \in P$ the subterm at occurence p in the term t with the term t'_p. A term $t \in \mathcal{T}(\mathrm{sig})$ is *linear* $\underline{\text{:iff}}$ $\forall p, q \in \mathcal{O}(t)$: $(t/p{=}t/q{\in}\mathrm{V} \;\Rightarrow\; p{=}q)$; it is *reducible* w. r. t. a relation $\Longrightarrow$ $\underline{\text{:iff}}$ $t \in \mathrm{dom}(\Longrightarrow)$.

2.3 Substitutions

The set of *substitutions* from a variable-system $X = (X_s)_{s \in \mathrm{S}}$ to an S-sorted family of sets $T = (T_s)_{s \in \mathrm{S}}$ is defined to be

$$\mathcal{SUB}(X,T) := \{\ \sigma : X \to T \mid \forall s \in \mathrm{S} : \forall x \in X_s : \sigma(x) \in T_s\ \}\ .$$

Important sets of substitutions are $\mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$ and $\mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{cons}))$. For the application of the homomorphic extension of a substitution $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$ to a term $t \in \mathcal{T}(\mathrm{sig})$ we use postfix-notation: $t\sigma$. Notice that for $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{cons}))$: $\mathcal{V}(t\sigma) = \emptyset$ [4].

We will use separations instead of the usual phrase "we always assume ... to have no variables in common". The set of *separations for* X, Y is defined for $X, Y \subseteq \mathrm{V}$ by:

$$\mathrm{Sep}(X,Y) \ := \ \{\ \xi \in \mathcal{SUB}(\mathrm{V},\mathrm{V}) \mid \xi \text{ bijective on V} \ \wedge\ \xi[X] \cap Y = \emptyset\ \}$$

For finite X, Y we always have $\mathrm{Sep}(X,Y) \neq \emptyset$ [5] and w. l. o. g.[6] we think '$\min \mathrm{Sep}(X,Y)$' to denote some element chosen from $\mathrm{Sep}(X,Y)$, because for our purposes all its elements are equivalent.

The set of *most general unifiers for* s, t is defined for $s, t \in \mathcal{T}(\mathrm{sig})$ by:

$$\begin{aligned}\mathrm{Mgu}(s,t) \ := \ \{\ & \sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})) \mid s\sigma = t\sigma \wedge {}^{7} \\ & \forall \mu \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})) : (s\mu = t\mu \ \Rightarrow\ \exists \tau \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})) : \sigma\tau = \mu)\ \}\end{aligned}$$

For $\exists \mu \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})) : s\mu = t\mu$ we always have $\mathrm{Mgu}(s,t) \neq \emptyset$ and w. l. o. g. we think '$\min \mathrm{Mgu}(s,t)$' to denote some element chosen from $\mathrm{Mgu}(s,t)$, because its elements are equivalent modulo variable-renaming.

[4] This is an application of the above-mentioned abuse of notation for families. A proper notation would be $\mathcal{V}(t\sigma) = (\emptyset)_{s \in \mathrm{S}}$

[5] because $\forall s \in \mathrm{S} : |\mathrm{V}_s| \notin \mathbb{N}$

[6] without loss of generality

[7] We could enlarge the conjunction by $\mathcal{V}(\sigma[\mathcal{V}(s,t)]) \subseteq \mathcal{V}(s,t)$ and $\sigma\sigma = \sigma$.

2.4 Orderings and other (binary) Relations

By an *irreflexive ordering* $<$ we mean an irreflexive and transitive relation, sometimes called "strict partial ordering" by other authors. As with all our asymmetric relation symbols: $a > b$ $\underline{:\text{iff}}$ $b < a$. Speaking of an *ordering* we always think of an irreflexive ordering. A relation R is *A-reflexive* iff $\mathrm{id}|_A \subseteq R$. Simply speaking of a *reflexive* relation we mean the biggest A that is appropriate in the local context. A *quasi-ordering* $\lesssim$ *on* A is an A-reflexive and transitive relation. An *equivalence on* A is an A-reflexive, symmetric, and transitive relation. The *equivalence* $\approx$ (on A) *of a quasi-ordering* $\lesssim$ (on A) is $\lesssim \cap \gtrsim$. The *ordering* $<$ *of a quasi-ordering* $\lesssim$ is $\lesssim \setminus \gtrsim$. A *reflexive ordering* $\leq$ *on* A is an A-reflexive, antisymmetric, and transitive relation. The *ordering* $<$ *of a reflexive ordering* $\leq$ is $\leq \setminus \mathrm{id}$. The *$A$-reflexive ordering* $\leq$ *of an ordering* $<$ is $< \cup\, \mathrm{id}|_A$.

A relation R on $\mathcal{T}(\mathrm{sig})$ is called:

sort-invariant $\underline{:\text{iff}}$ $\forall(t,t') \in R : \exists s \in \mathrm{S} :\ t,t' \in \mathcal{T}(\mathrm{sig})_s$

stable[8] *(w. r. t. substitution)* $\underline{:\text{iff}}$ $\forall(t,t') \in R \colon \forall\sigma \in \mathcal{SUB}(\mathrm{V},\mathcal{T}(\mathrm{sig}))\colon (t\sigma, t'\sigma) \in R$

monotonic (w. r. t. replacement)
$\underline{:\text{iff}}$ $\forall(t',t'') \in R : \forall t \in \mathcal{T}(\mathrm{sig}) : \forall p \in \mathcal{O}(t) : \forall s \in \mathrm{S} :$
$$(t/p, t', t'' \in \mathcal{T}(\mathrm{sig})_s \ \Rightarrow\ (t[\,p \leftarrow t'\,], t[\,p \leftarrow t''\,]) \in R)$$

sufficiently complete (w. r. t. $\mathcal{GT}(\mathrm{cons})$*)* $\underline{:\text{iff}}$ $\forall t \in \mathcal{GT}(\mathrm{sig})\colon \exists t' \in \mathcal{GT}(\mathrm{cons})\colon (t,t') \in \mathrm{R}$
ground ... $\underline{:\text{iff}}$ $R \cap (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))$ is ...

A relation R is called:

total on A $\underline{:\text{iff}}$ $A \times A \ \subseteq\ (\mathrm{id}|_A \cup R \cup R^{-1})$
noetherian $\underline{:\text{iff}}$ there is no $a : \mathbb{N} \to \mathrm{field}(R)$ with $\forall i \in \mathbb{N} : (a_i, a_{i+1}) \in R$
normalizing $\underline{:\text{iff}}$ $\forall t : \exists t' : ((t,t') \in R^{\circledast} \ \wedge\ t' \notin \mathrm{dom}(R))$

An ordering $<$ or $>$ is called *well-founded* $\underline{:\text{iff}}$ $>$ is noetherian. A quasi-ordering or a reflexive ordering is called *well-founded* $\underline{:\text{iff}}$ its ordering is well-founded.

A *congruence* is a monotonic equivalence.

A *reduction ordering* on $\mathcal{T}(\mathrm{sig})$ is a monotonic, stable, and well-founded ordering.

The *(proper) subterm ordering* $\lhd_{\mathrm{ST}}$ is the stable and well-founded ordering defined by:
$$t \unlhd_{\mathrm{ST}} t' \ \underline{:\text{iff}}\ \exists p \in \mathcal{O}(t') : t = t'/p$$

A *simplification ordering* is a reduction ordering containing $\lhd_{\mathrm{ST}}$.

For further information on orderings see [11].

[8] Similarly a predicate P on $\mathcal{T}(\mathrm{sig})$ is stable $\underline{:\text{iff}}$ $\forall t \in P \colon \forall\sigma \in \mathcal{SUB}(\mathrm{V},\mathcal{T}(\mathrm{sig}))\colon t\sigma \in P$

2.5 $\Longrightarrow$, Confluence, and Church-Rosser-Property

The symmetric closure of a relation $\Longrightarrow$ will be denoted by $\Longleftrightarrow$.
Two terms v, w are called *joinable w. r. t.* $\Longrightarrow$ <u>:iff</u> $v \downarrow w$ <u>:iff</u> $v \overset{\circledast}{\Longrightarrow} \circ \overset{\circledast}{\Longleftarrow} w$.
A relation $\Longrightarrow$ is called *confluent below u* <u>:iff</u> $\forall v, w : ((v \overset{\oplus}{\Longleftarrow} u \overset{\oplus}{\Longrightarrow} w) \;\Rightarrow\; (v \downarrow w))$.
A relation $\Longrightarrow$ is called *locally confluent below u* <u>:iff</u> $\forall v, w : ((v \Longleftarrow u \Longrightarrow w) \;\Rightarrow\; (v \downarrow w))$;
it is called *[locally] confluent* <u>:iff</u> it is [locally] confluent below all u.
A relation $\Longrightarrow$ is said to have the *Church-Rosser-property* <u>:iff</u> $\overset{\circledast}{\Longleftrightarrow} \subseteq \downarrow$.

Lemma 2.2 *A noetherian relation $\Longrightarrow$ is confluent <u>iff</u> it is locally confluent.*

Lemma 2.3 *A relation $\Longrightarrow$ is confluent <u>iff</u> it has the Church-Rosser-property.*

2.6 Algebras

Following the style of [14] we define a sig-algebra $\mathcal{A}$ over the signature $\mathrm{sig} = (\mathrm{F}, \mathrm{S}, \alpha)$ to be a function $\mathcal{A} : (\mathrm{F} \uplus \mathrm{S}) \to \mathcal{U}$ with $\forall s \in \mathrm{S} : \mathcal{A}(s) \neq \emptyset$ and

$$f^{\mathcal{A}} : \mathcal{A}(s_0) \times \cdots \times \mathcal{A}(s_{n-1}) \to \mathcal{A}(s_n) \qquad \text{for } f \in \mathrm{F} \text{ with } \alpha(f) = s_0 \ldots s_{n-1}s_n.$$

We will write $f^{\mathcal{A}}$ instead of $\mathcal{A}(f)$ for $f \in \mathrm{F}$. $\mathcal{A}(s)$ will be called the universe of $\mathcal{A}$ for the sort s.

A sig-algebra $\mathcal{A}$ is called *trivial* <u>:iff</u> $\forall s \in \mathrm{S} : |\mathcal{A}(\mathrm{s})| = 1$. The differences between two trivial sig-algebras not being too interesting we speak of *the* trivial sig-algebra if we mean any.

An *$\mathcal{A}$-valuation b (of* V*)* is an element of $\mathcal{SUB}(\mathrm{V}, (\mathcal{A}(s))_{s \in \mathrm{S}})$[9]. The evaluation function $\mathcal{A}_b$ is recursively defined as follows:

$$\begin{array}{rcll} \mathcal{A}_b(x) &=& b(x) & \text{for } x \in \mathrm{V} \\ \mathcal{A}_b(f t_0 \ldots t_{n-1}) &=& f^{\mathcal{A}}(\mathcal{A}_b(t_0), \ldots, \mathcal{A}_b(t_{n-1})) & \text{for } f \in \mathrm{F};\ t_0, \ldots, t_{n-1} \in \mathcal{T}(\mathrm{sig}) \end{array}$$

For getting acquainted with this notation, here is a well-known lemma:

Lemma 2.4 (Substitution-Lemma) *Let $\mathcal{A}$ be a* sig*-algebra and b an $\mathcal{A}$-valuation of* V*. For $t \in \mathcal{T}(\mathrm{sig})$ and $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$:*

$$\mathcal{A}_b(t\sigma) = \mathcal{A}_{\sigma \circ \mathcal{A}_b}(t)$$

By abuse of notation we sometimes use $\mathcal{T}(\mathrm{sig})$ to denote the *term algebra* over sig/V instead of the family of terms. This term algebra $\mathcal{A}$ has $\mathcal{T}(\mathrm{sig})_s$ as the universe for each sort s and works on function symbols as follows:

$$\begin{array}{rcll} f^{\mathcal{A}}(t_0, \ldots, t_{n-1}) &=& f t_0 \ldots t_{n-1} & \text{for } f \in \mathrm{F} \text{ with } \alpha(f) = s_0 \ldots s_{n-1}s_n \\ &&& \text{and } \forall i < n :\ t_i \in \mathcal{T}(\mathrm{sig})_{s_i} \end{array}$$

Similarly, we use $\mathcal{GT}(\mathrm{sig})$ for the *ground term algebra* over sig instead of the family of ground terms.

A sig-algebra $\mathcal{A}$ is called cons*-term-generated* <u>:iff</u>

$$\forall s \in \mathrm{S} : \forall a \in \mathcal{A}(s) : \exists t \in \mathcal{GT}(\mathrm{cons})_s :\ \mathcal{A}(t) = a$$

[9] One would not call b a substitution, however, unless $\mathcal{A}$ is a term algebra.

3 Motivation

Unconditional Equations

The use of a finite set R of unconditional equations for computing a $<$-minimal R-congruent output term for a given input term is a very well-known and easy procedure: One just substitutes the occurrence (called the *redex*) of one side of an instantiated equation in the input term by the other side of this instantiated equation, provided that this leads to a term (called the *reduct*) that is $<$-smaller than the input term; and repeats this procedure as long as possible. It is generally accepted that $<$ should be a reduction ordering, because then we can try to orient the equations into *rules* that are elements of $>$ and thereby guarantee the reduct always to be $<$-smaller than the input term if one substitutes the *left-hand side* of an instantiated rule by its *right-hand side*.

Positive-Conditional Equations

When changing from unconditional to positive-conditional equations[10] that only have to hold if their condition holds[11], the class of algebras satisfying R still contains an initial algebra, whereas the computation of a $<$-minimal R-congruent term gets more difficult: The replacement of one side of an instantiated rule by the other is uncertain until the condition of the instantiated rule has been shown to hold in the initial algebra of R. Now, while the applicability of a rule to a given term is semi-decidable[12], it is not surprising that it is not decidable in general — even if all variables of a rule occur in its left-hand side, the reduction relation is noetherian and confluent, and the term is a ground term.[13]

Thus, for guaranteeing decidability of applicability of a rule to a given term (and decidability of reducibility of a given term[14]), one has to require the terms in the condition of a rule

1. to be somehow smaller than the left-hand side of the rule

and

2. to contain only variables that also occur in the left-hand side of the rule.

The former restriction limits the depth of the "check-condition"-calls; the latter abandons enumerating all possible solutions for the "extra"[16] variables in the condition. The most appropriate and least restrictive way of telling the terms in the condition to be smaller than the left-hand side of the rule seems to be by usage of the same reduction ordering $<$ again. We actually use an additional stable and well-founded ordering (containing $<$ and $\lhd_{\mathrm{ST}}$) for this step. This is explained in section 7.2, where we will introduce a pair of orderings, called a *termination-pair*, that perfectly fits the task, just as the reduction ordering does in the unconditional case.

[10] By a positive-conditional equation we mean an equation that has associated with it a condition that consists of a (possibly empty) conjunction of positive (i. e. not negated) equations. For general information on positive-conditional equations see Kaplan[22].

[11] i. e. $\forall \vec{x} : (\mathrm{condition}(\vec{x}) \;\Rightarrow\; \mathrm{equation}(\vec{x}))$

[12] by breadth-first search

[13] Cf. section 7.1

[14] which implies[15] the computability of normal forms for noetherian reduction relations

[15] Cf. Lemma 7.3

[16] i. e. not occurring in the left-hand side of the rule

Positive/Negative-Conditional Equations

Further problems occur when one allows negated equations among the equations of the conjunction that forms the condition of the equation. Kaplan[23] defines a positive equation in the condition to hold <u>:iff</u> its terms have a common reduct (w. r. t. $\overset{\circledast}{\Longrightarrow}$), and a negated equation to hold <u>:iff</u> its terms have no common reduct. If the resulting reduction relation is confluent and the rules are compatible with some termination-pair then its congruence closure is minimal (but not a minimum!) w. r. t. set-inclusion among the congruence relations whose factor algebra (w. r. t. $\mathcal{GT}$(sig)) is a model[17] of R.

Despite of the lack of an initial model even in this restricted case, positive/negative-conditional equations are really necessary for convenient specification, as can be seen by the following example with PROLOG-like syntax.

Example 3.1
(continuing Example 2.1)

R:

$$
\begin{array}{lcllll}
- \; x \; \mathsf{0} & = & x & & \\
- \; \mathsf{s} \; x \; \mathsf{s} \; y & = & - \; x \; y & & \\
 & & & & \\
\mathsf{memberp} \; x \; \mathsf{nil} & = & \mathsf{false} & & \\
\mathsf{memberp} \; x \; \mathsf{cons} \; y \; l & = & \mathsf{true} & \longleftarrow & x = y \\
\mathsf{memberp} \; x \; \mathsf{cons} \; y \; l & = & \mathsf{memberp} \; x \; l & \longleftarrow & x \neq y \\
 & & & & \\
\mathsf{subsetp} \; \mathsf{nil} \; l & = & \mathsf{true} & & \\
\mathsf{subsetp} \; \mathsf{cons} \; x \; k \; l & = & \mathsf{false} & \longleftarrow & \mathsf{memberp} \; x \; l \neq \mathsf{true} \\
\mathsf{subsetp} \; \mathsf{cons} \; x \; k \; l & = & \mathsf{subsetp} \; k \; l & \longleftarrow & \mathsf{memberp} \; x \; l = \mathsf{true}
\end{array}
$$

[17] taking the conditional equations as universally quantified formulas of first order equational logic (as indicated in footnote 11) and using standard algebra model semantics

The Importance of Confluence

In this section we explain why confluence is essential for reduction with positive/negative-conditional rules.

Firstly (even without negative equations), confluence is needed for the completeness of testing semantic equality of two condition terms by looking for a common reduct[18]. This means: We need confluence for the congruence defined in Kaplan[23] to yield a model of R. Secondly, it is needed for guaranteeing the congruence to be minimal, as can be seen from:

Example 3.2

Let a, b, c, d, e be constants of the same sort, $a > b > c > d > e$.

$$\begin{array}{lll} \mathrm{R}: & c = d & \\ & c = e & \\ & a = b & \longleftarrow \quad e \neq d \end{array}$$

Then, for the reduction relation of Kaplan[23]:

$\Longrightarrow \; = \; \{ \; (c, d), \; (c, e), \; (a, b) \; \}$ *, whose congruence closure* $\overset{\circledast}{\Longleftrightarrow}$ *is not minimal.*

While confluence can be dropped for merely positive conditional equations by testing for congruence instead for a common reduct of two condition terms, it is worse for positive/negative-conditional equations: It does not suffice to test non-congruence for inequality of two condition terms if confluence is not provided, as can be seen from the following example:

Example 3.3

Let the signature and the ordering be as in the previous example.

$$\begin{array}{lll} \mathrm{R}: & c = d & \longleftarrow \quad d \neq e \\ & c = e & \end{array}$$

Any congruence yielding a model of R must contain (d, e): If it did not, it would contain (c, d) and (c, e), and hence (d, e). Therefore, no matter which congruence we actually use for condition-testing, the test of $d \neq e$ with such a (model-yielding) congruence will always fail, so that we cannot establish $c \overset{\circledast}{\Longleftrightarrow} d$ by testing the condition of the first rule. But R has the minimum model $c = d = e$, which cannot be obtained by the simple technique of condition-testing anymore, but only by paramodulation and factoring instead, which in our opinion is too complicated a procedure for establishing just a simple reduction step.

By this we conclude that in case of negative equations in the condition, computation of a correct reduct by the technique of condition-testing is only possible if confluence is provided.

---

[18] w. r. t. the reflexive and transitive closure

Criticism of Kaplan[23]

The major shortcoming of the reduction relation in [23], however, is (as noted above) that its congruence closure is not a minimum[19] but only minimal[20] among the congruences yielding a model of R. This means that there might be reductions $s \Longrightarrow t$ with $s = t$ not holding in all models logically specified by R.

Kaplan[23] argues as follows:

By writing $c = d \longleftarrow d \neq e$ instead of the logically equivalent $c = d \lor d = e$ the specifier adds some "operational" information to control the choice of the intended minimal congruence '$c = d$' from the congruences yielding a model of R ('$c = d$', '$d = e$', and '$c = d = e$').

But writing $c = d \lor d = e$ in the form of $c = d \longleftarrow d \neq e$ is actually motivated by the ordering aspect and not by the specifier's[21] intention which of all minimal models to choose. What's worse is that semantics is given by control and not expressible without; thereby violating the paradigm of separation of logic and control: It is not a *logical* specification (suitable for computation) anymore (as it was the case with positive-conditional equations), but a program with none but operational semantics. Therefore we lose all the nice aspects of the pure logic view. To name one: The monotony of logic is lost: If we, e. g., complete the specification of a partially specified function[22] in a proper way[23], we might destroy some reductions and congruences[24] that were possible before:

Example 3.4 (continuing Example 3.1)
We have $\mathsf{memberp}\ \mathsf{0}\ \mathsf{cons}\ -\ \mathsf{0}\ \mathsf{s}\ \mathsf{0}\ \mathsf{nil} \overset{\circledast}{\Longrightarrow} \mathsf{false}$ *which no longer holds after adding the rule* $-\ \mathsf{0}\ \mathsf{s}\ x\ =\ \mathsf{0}$.

By this example we can recognize a similar problem: Reduction of non-ground terms isn't of any use because the reduction relation isn't stable:

Example 3.5 *As* x *does not reduce to* $\mathsf{0}$ *, one might say*

$$\mathsf{memberp}\ \mathsf{0}\ \mathsf{cons}\ x\ \mathsf{nil} \overset{\circledast}{\Longrightarrow} \mathsf{false}\ .$$

But for $x \mapsto \mathsf{0}$ *this is nonsense.*

Looking for Remedy

One could think that the problem of a minimal congruence not being a minimum hardly arises or is avoidable by convenient purely syntactic restrictions on the defining rules. The following example will exhibit that such restrictions for the congruence of [23] cannnot be reasonable in practice of specification:

[19] i. e. being smaller than anything else
[20] i. e. there is nothing smaller
[21] Who is very likely to be unable to keep track of all consequences of his pieces of "operational" information
[22] And partially specified functions are very useful in expressing exactly what is required by the specifier!
[23] i. e. not confusing different constructor terms
[24] by this we mean elements of $\overset{\circledast}{\Longleftrightarrow}$

Example 3.6 (continuing Example 3.1)
We exclude[25] *the function symbols* subsetp, 0, s, *and* − *(together with their respective rules), and enrich the signature with the following two constants:* $\alpha(\mathsf{a}) = \alpha(\mathsf{b}) = \mathsf{int}$. *The reduction relation* $\Longrightarrow$ *of [23] is confluent*[26] *and noetherian*[27] *in this case.*

Consider the following two congruence relations on ground terms, given by their congruence classes:

$\overset{\circledast}{\Longleftrightarrow}$:
$\{\mathsf{a}\}$
$\{\mathsf{b}\}$
$\{\mathsf{false}\} \cup \{\mathsf{memberp}\ x\ l \ |(x \in \{\mathsf{a},\mathsf{b}\} \ \wedge \ (x \textit{ doesn't occur in } l))\}$
$\{\mathsf{true}\} \cup \{\mathsf{memberp}\ x\ l \ |(x \in \{\mathsf{a},\mathsf{b}\} \ \wedge \ (x \textit{ does occur in } l))\}$
$\{\mathsf{nil}\}$
$\{\mathsf{cons\ a\ nil}\}$
$\{\mathsf{cons\ b\ nil}\}$
$\{\mathsf{cons\ a\ cons\ a\ nil}\}$
$\{\mathsf{cons\ a\ cons\ b\ nil}\}$
$\vdots$

$\sim$:
$\{\ \mathsf{a},\ \mathsf{b}\ \}$
$\{\ \mathsf{false},\ \mathsf{memberp\ a\ nil},\ \mathsf{memberp\ b\ nil}\ \}$
$\{\mathsf{true}\} \cup \{\mathsf{memberp}\ x\ l \ |(x \in \{\mathsf{a},\mathsf{b}\} \ \wedge \ l \neq \mathsf{nil})\}$
$\{\mathsf{nil}\}$
$\{\ \mathsf{cons\ a\ nil},\ \mathsf{cons\ b\ nil}\ \}$
$\{\mathsf{cons}\ x\ \mathsf{cons}\ y\ \mathsf{nil} \mid x, y \in \{\mathsf{a},\mathsf{b}\}\ \}$
$\vdots$

Now, both $\overset{\circledast}{\Longleftrightarrow}$ *and* $\sim$ *yield a model of* R. *By* $a \sim b$; $a \overset{\circledast}{\not\Longleftrightarrow} b$; *we know that* $\sim$ *is not a minimum. By* $\mathsf{memberp\ a\ cons\ b\ nil} \overset{\circledast}{\Longleftrightarrow} \mathsf{false}$ *and* $\mathsf{memberp\ a\ cons\ b\ nil} \not\sim \mathsf{false}$ *we know that* $\overset{\circledast}{\Longleftrightarrow}$ *isn't a minimum either. But both* $\overset{\circledast}{\Longleftrightarrow}$ *and* $\sim$ *are minimal among the congruences that yield a model of* R. *Hence their intersection does not yield a model of* R.

Notice that the example is really a standard specification example and not a sophisticated one. Therefore, restrictions on the structure of the rules for avoiding minimal non-minimum models would necessarily forbid such basic and essential specifications as the memberp-example above.

Thus, we have to choose between $\mathsf{a} \neq \mathsf{b}$ and $\mathsf{memberp\ a\ cons\ b\ nil} \neq \mathsf{false}$. As $\overset{\circledast}{\Longleftrightarrow}$ is somehow more appealing than $\sim$, one may argue that $\mathsf{a} \neq \mathsf{b}$ is somewhat more important than $\mathsf{memberp\ a\ cons\ b\ nil} \neq \mathsf{false}$ by stating a, b to be constructors and thinking freeness of constructors to be more important than that of non-constructors. But this treatment does not solve the problem in general: If

1. a or b is changed into a (composite) non-constructor term,

or

2. memberp is stated to be a constructor symbol too,

then the very same problem arises again.

[25] It's not that they make trouble. We just omit them to make the example simpler.
[26] There are no feasible critical pairs
[27] Use the lexicographic path ordering given by $\mathsf{memberp} \succ \mathsf{true}, \mathsf{false}$

Our Solution

Now, while the simple attempt above fails, the intended bias towards freeness of constructor terms can be achieved the following way:

1. Adding condition literals (cf. sect. 4) expressing *definedness* for all terms of negative equations in the condition. A term is *defined* :iff it has a congruent constructor ground term. For our example above this means that the last $\mathsf{memberp}$-rule is not applicable if a or b is undefined, thereby avoiding the problem of (1) above.
2. Forcing rules whose left-hand sides are constructor terms to have no negative equations in their conditions and to be constructor-preserving[28]. For our example above this means that $\mathsf{memberp}$ cannot be a constructor symbol, thereby avoiding the problem of (2) above.

(2) is purely syntactic and not very restrictive in practice as it only limits congruences between constructor terms (and this even less restrictively than usual). (1) is not a usage of control information as before. It just means that '$\neq$' is syntactically restricted to defined terms. Undefined terms are due to partially specified functions or incomplete knowledge about the model world. They are often thought to be equal to some unknown constructor ground terms: Kapur&Musser[25, 26] consider those congruences which are maximally enlarged by random identification of undefined terms with constructor ground terms, as long as this identification does not make two distinct constructor ground terms equal. The intended congruence is then the intersection of all those maximally enlarged congruences. In [25] the maximal congruences are allowed to have some undefined terms left; this causes the problem that one cannot describe the intended congruence by model semantics[29]. Therefore in [26] the intersection is done only over those congruences that have no undefined terms left: These congruences can be easily described in terms of model semantics: A model $\mathcal{A}$ is additionally[30] required to satisfy the following: Let $\overset{\circledast}{\Longleftrightarrow}$ denote the initial congruence[31] of R and c its canonical[32] cons-epimorphism from $\mathcal{GT}(\text{cons})$ to $\mathcal{GT}(\text{cons})/\overset{\circledast}{\Longleftrightarrow}$. Now the unique cons-homomorphism[33] h from $\mathcal{GT}(\text{cons})/\overset{\circledast}{\Longleftrightarrow}$ to $\mathcal{A}$ given by $c \circ h = \mathcal{A}$ is required to be an isomorphism. A third way of removing the undefined terms is to require h to be epimorphic instead of isomorphic, i. e. $\mathcal{A}$ is additionally[30] required to be cons-term-generated. While the theory of the last two attempts is very beautiful[34], the resulting congruences may be very difficult to understand: One needs a very sophisticated way of argumentation for showing two terms equal — even for some very simple examples.

[28] i. e.: all terms in such a "constructor rule" are constructor terms and all variables of a constructor rule occur in its left-hand side

[29] Of course, this is tried in [25]. But their "inductive model" (which is defined to be a model whose proper epimorphic images are no models) is a nasty thing: Normally, a model uses to keep being a model when one throws away some equations of the specification, thereby establishing the monotony of logic. The "inductive models" do not have this property. To see this take $\mathrm{C} = \{\mathsf{false}, \mathsf{true}, \mathsf{0}\}$; $\mathrm{D} = \{\mathsf{s}, \mathsf{zerop}\}$; $\mathrm{R} = \{\ \mathsf{zerop}\,\mathsf{0} = \mathsf{true},\ \ \mathsf{zerop}\,\mathsf{s}\,x = \mathsf{false}\ \}$. Now the following $\mathcal{A}$ is an "inductive model" for R but not for $\emptyset$ (where we need $|\mathcal{A}(\mathsf{int})| = 1$): $\mathcal{A}(\mathsf{bool}) = \{\mathsf{FALSE}, \mathsf{TRUE}\}$; $\mathcal{A}(\mathsf{int}) = \{0, 1\}$; $\mathsf{true}^{\mathcal{A}} = \mathsf{TRUE}$; $\mathsf{false}^{\mathcal{A}} = \mathsf{FALSE}$; $\mathsf{0}^{\mathcal{A}} = 0$; $\mathsf{s}^{\mathcal{A}}(x) = 1$; $\mathsf{zerop}^{\mathcal{A}}(0) = \mathsf{TRUE}$; $\mathsf{zerop}^{\mathcal{A}}(1) = \mathsf{FALSE}$.

We can also see by this that we indeed have no monotonic logic here: $\emptyset \vdash \mathsf{0} = \mathsf{s}\,\mathsf{0}$; but (as seen by $\mathcal{A}$): $\mathrm{R} \not\vdash \mathsf{0} = \mathsf{s}\,\mathsf{0}$.

[30] besides making true the universally quantified equations of R

[31] which exists because [25, 26] consider unconditional equations only

[32] sometimes called *natural* instead of canonical; mapping each term of $\mathcal{GT}(\text{cons})$ to its congruence class.

[33] given by the Homomorphism-Theorem

[34] We resist the temptation to include it here.

Based on this tradition of thinking undefined terms to be possibly equal to constructor ground terms, the above item (1) of our approach can be justified the following way: If two terms can be shown equal by $\overset{\circledast}{\Longleftrightarrow}$, they will stay equal even if an undefined term will be identified with a constructor ground term later on. Therefore, equality is a safe thing. On the other hand might an undefined term become equal (w. r. t. $\overset{\circledast}{\Longleftrightarrow}$) to a previously unequal term when identifying an undefined term with a constructor ground term. Thus, we had better be cautious: We should not pretend to be able to distinguish something undefined from anything else (as the former might in the sequel be defined to be the latter).

In the following sections we will show that by the little changes of (1),(2) we get a straightforward reduction relation $\Longrightarrow$ that has the following advantages (compared to the one of Kaplan[23]) (cf. sect. 5):

1. Its congruence closure $\overset{\circledast}{\Longleftrightarrow}$ yields a minimal model that is (beyond that) the minimum of all models of R that do not identify more constructor ground terms than necessary, provided (as before[35]) that $\Longrightarrow$ is confluent.
2. $\Longrightarrow$ is monotonic w. r. t. the addition of new rules that do not have left-hand sides which are old constructor terms.
3. $\Longrightarrow$ is stable when defined also on non-ground terms.

As shown in the examples above, the reduction relation of [23] has none of these properties. We will now revisit these examples to illustrate how our restrictions solve their problems.

1. (Example 3.6). If a and b are constructor terms, then $\overset{\circledast}{\Longleftrightarrow}$ becomes the *minimum* among those congruences which do not identify more constructor ground terms than necessary. Contrariwise, if a or b is undefined, then the intersection of $\overset{\circledast}{\Longleftrightarrow}$ and $\sim$ becomes a model of R because the last $\mathsf{memberp}$-rule now reads "$\mathsf{memberp}\ x\ \mathsf{cons}\ y\ l = \mathsf{memberp}\ x\ l \longleftarrow x \neq y,\ \mathrm{Def}\,x,\ \mathrm{Def}\,y$"; thus, $\mathsf{memberp\ a\ cons\ b\ nil}$ is neither true nor false but undefined instead.
2. (Example 3.4). We do not have $\mathsf{memberp\ 0\ cons\ -\ 0\ s\ 0\ nil} \overset{\circledast}{\Longrightarrow} \mathsf{false}$ anymore: $\mathsf{memberp\ 0\ cons\ -\ 0\ s\ 0\ nil}$ is irreducible because $\mathsf{-\ 0\ s\ 0}$ is undefined.
3. (Example 3.5). As x is undefined: $\mathsf{memberp\ 0\ cons}\ x\ \mathsf{nil} \overset{\circledast}{\not\Longrightarrow} \mathsf{false}$.

Furthermore, notice that if our reduction relation is confluent and there aren't any undefined ground terms[36], there is no difference between our ground reduction relation and that of Kaplan[23]. Therefore in this important case we offer semantics for the reduction relation of [23] not using any control information.

Finally, we are not only able to specify our semantics without using control information, but also able to remove the control aspect of requiring the rules to be compatible with a termination-pair for admissibility of a specification. As a consequence, our reduction relation does not need to be noetherian or restricted to ground terms.

[35] Kaplan[23] additionally has to require $\Longrightarrow$ to be compatible with a termination-pair and thereby to be noetherian.

[36] i. e. iff our $\overset{\circledast}{\Longrightarrow}$ is *sufficiently complete*.

4 Syntax and Semantics of the Specification Language

Definition 4.1 (Syntax of CRS) *A (positive/negative-)*conditional rule system *(*CRS*)* R *over* sig*,* cons *and* V *is a finite subset of the* set of rules $\mathcal{RUL}(\mathrm{sig},\mathrm{cons})$ *over* sig*,* cons *and* V*, that will be defined in Definition 5.1. The only thing we have to know about it now is:* $\mathcal{RUL}(\mathrm{sig},\mathrm{cons}) \subseteq \mathrm{DEq}(\mathrm{sig}) \times (\mathcal{LIT}(\mathrm{sig}))^*$ *, where* $\mathcal{LIT}(\mathrm{sig})$ *is the set of* condition literals *over* sig*:*

$$\begin{array}{rcll}
\mathcal{LIT}(\mathrm{sig}) & := & (\quad \mathrm{Eq}(\mathrm{sig}) \times \{=\ \}\ ) \cup & \textit{"positive equations"} \\
 & & (\quad \mathrm{Eq}(\mathrm{sig}) \times \{\neq\ \}\ ) \cup & \textit{"negative equations"} \\
 & & (\ \mathrm{SMul}(\mathcal{T}(\mathrm{sig})) \times \{\mathrm{Def}\}\ ) & \textit{"definedness predicates"} \\
\mathrm{Eq}(\mathrm{sig}) & := & \{\ \langle t,t'\rangle \mid \exists s \in \mathrm{S} : t,t' \in \mathcal{T}(\mathrm{sig})_s\ \} & \textit{"set of equations[37]"} \\
\mathrm{DEq}(\mathrm{sig}) & := & \{\ (t,t') \mid \exists s \in \mathrm{S} : t,t' \in \mathcal{T}(\mathrm{sig})_s\ \} & \textit{"set of directed equations"} \\
\mathrm{SMul}(X) & := & \{\ \langle t\rangle \mid t \in X\ \} & \textit{"set of singleton multisets"}
\end{array}$$

A rule *with an empty* condition $((l,r),\emptyset) \in \mathcal{RUL}(\mathrm{sig},\mathrm{cons})$ *will be written as* $l{=}r$*. A rule* $((l,r),C)$ *from* $\mathcal{RUL}(\mathrm{sig},\mathrm{cons})$ *with condition* C *will be written* $l{=}r \longleftarrow C$*. The condition literals will be written in the following way: '*$u{=}v$*' for* $(\langle u,v\rangle,=)$*, '*$u{\neq}v$*' for* $(\langle u,v\rangle,\neq)$*, '*$\mathrm{Def}\,u$*' for* $(\langle u\rangle,\mathrm{Def})$*. We call* l *the* left-hand side *and* r *the* right-hand side *of the rule* $((l,r),C)$*; the elements of the first position of the condition literals in* C *are called* condition terms *and their set is denoted by* $\mathcal{TERMS}(C)$*. The set of all left-hand sides of rules in* R *is denoted by* lhs(R)*.* R *is* left-linear *:iff all elements of* lhs(R) *are linear terms. A rule* $((l,r),C) \in \mathcal{RUL}(\mathrm{sig},\mathrm{cons})$ *is said to be* extra-variable free *:iff* $\mathcal{V}(\mathcal{TERMS}(C),r) \subseteq \mathcal{V}(l)$*.* R *is* extra-variable free *:iff all its rules* $((l,r),C) \in \mathrm{R}$ *are extra-variable free.*

A rule $((l,r),C)$ from $\mathcal{RUL}(\mathrm{sig},\mathrm{cons})$ expresses a universally quantified implication with the conjunction of the literals in C as the condition and with '$l = r$' as the conclusion[38]. The meaning of the meta-predicate symbols '=', '≠', and 'Def' is not open to interpretation. The fixed meaning of '=' is standard; '≠' is its negation. 'Def', however, is the "definedness" predicate which states that the evaluation of its argument belongs (with sort invariant) to the cons-term-generated cons-subalgebra[39] of $\mathcal{A}$. We speak of our new kind of model just as a "sig/cons-model" (without any further attributes), because if we removed the new meta-predicate symbol 'Def', we would just get the usual model concept of algebra; i. e., our sig/cons-model is an upward-compatible extension.

Definition 4.2 (Semantics of CRS) *Let* R *be a CRS over* sig*,* cons *and* V*; let* $\mathcal{A}$ *be a* sig*-algebra. Now* $\mathcal{A}$ *is a* sig/cons-model of R **:iff**

$$\forall((l,r),C) \in \mathrm{R} : \forall b\ \mathcal{A}\text{-valuation of } \mathrm{V} : \Big((C \textit{ is } \mathbf{true\ w.\ r.\ t.}\ \mathcal{A}_b) \ \Rightarrow\ \mathcal{A}_b(l) = \mathcal{A}_b(r)\Big)\ ,$$

where C *is* **true w. r. t.** $\mathcal{A}_b$ *:iff*

$$\forall s \in \mathrm{S}\colon \forall u,v \in \mathcal{T}(\mathrm{sig})_s : \begin{pmatrix} ((\ (u{=}v\ )\ \textit{in } C) \ \Rightarrow\ \mathcal{A}_b(u) = \mathcal{A}_b(v)\ ) & \wedge \\ ((\ (u{\neq}v\ )\ \textit{in } C) \ \Rightarrow\ \mathcal{A}_b(u) \neq \mathcal{A}_b(v)\ ) & \wedge \\ ((\ (\mathrm{Def}\,u)\ \textit{in } C) \ \Rightarrow\ \mathcal{A}_b(u) \in \mathcal{A}[\mathcal{GT}(\mathrm{cons})_s]\ ) & \end{pmatrix}$$

[37]where "$\langle u,\ldots\rangle$" denotes the multiset with the elements u, … — resembling the set notation $\{u,\ldots\}$

[38]Notice that we don't make any use of the fact that we have $(l,r) \in \mathrm{DEq}(\mathrm{sig})$ beyond $\langle l,r\rangle \in \mathrm{Eq}(\mathrm{sig})$ here; this ordering-property will be used for reduction only, and we have supplied our rules with it from the beginning for not having to change them afterwards.

[39] When speaking of a cons-subalgebra of a sig-algebra we mean a cons-subalgebra of the cons-algebra which is the restriction of the sig-algebra to cons.

As we have negative equations in our conditions, we cannot hope to get a minimum model because we can express things like ' $a = b \ \vee \ b = c$ ', which has the incomparable minimal models '$a = b \neq c$' and '$a \neq b = c$'. What we will get instead is a model that is the minimum of all models that do not identify more constructor ground terms than necessary.[40] For formally expressing this minimality-property, we need the following definition.

Definition 4.3

1. *Define* $\lesssim_{\mathrm{H}}$ *and* $\lesssim_{\mathrm{cons}}$ *as (proper class) relations on* sig*-algebras by (*$\mathcal{A}$*,* $\mathcal{B}$ sig*-algebras):*
   $\mathcal{A} \lesssim_{\mathrm{H}} \mathcal{B}$ **:iff** *there is a* sig*-homomorphism from* $\mathcal{A}$ *to* $\mathcal{B}$*.*
   $\mathcal{A} \lesssim_{\mathrm{cons}} \mathcal{B}$ **:iff** *there is a* cons*-homomorphism from the* cons*-subalgebra*[41] *of* $\mathcal{A}$ *that has the universes* $\mathcal{A}[\mathcal{GT}(\mathrm{cons})_s]$ *to the analogous subalgebra of* $\mathcal{B}$*.*

   We trivially get $\lesssim_{\mathrm{H}} \ \subseteq \ \lesssim_{\mathrm{cons}}$ *(by restriction of the homomorphism); and* $\lesssim_{\mathrm{H}}$*,* $\lesssim_{\mathrm{cons}}$ *are quasi-orderings. The corresponding equivalences, orderings, and reflexive orderings will be denoted by* $\approx$*,* $<$*,* $\leq$*, resp., with the corresponding subscript.*

2. *A* sig*-algebra* $\mathcal{A}$ *will be called a* minimum model *(or else a* constructor-minimum model*) of a CRS* R *over* sig/cons/V **:iff** $\mathcal{A}$ *is a* $\lesssim_{\mathrm{H}}$*-minimum (or else* $\lesssim_{\mathrm{cons}}$*) of the class of all* sig/cons*-models of* R*.*
   Similarly, a sig*-algebra* $\mathcal{A}$ *will be called a* minimal model *(or else a* constructor-minimal model*) of a CRS* R *over* sig/cons/V **:iff** $\mathcal{A}$ *is a* sig/cons*-model of* R *and there is no* sig/cons*-model* $\mathcal{B}$ *of* R *with* $\mathcal{B} <_{\mathrm{H}} \mathcal{A}$ *(or else* $\mathcal{B} <_{\mathrm{cons}} \mathcal{A}$ *).*

The following lemma tells us that, considering minimum models, we can think in terms of congruences on $\mathcal{GT}(\mathrm{sig})$ instead of algebras:

Lemma 4.4 *Let* $\mathcal{B}$ *be a* sig/cons*-model of the CRS* R *over* sig/cons/V*. Define* $\sim$ *by* $(s,t \in \mathcal{GT}(\mathrm{sig}))$*:* $s \sim t$ *:iff* $\mathcal{B}(s) = \mathcal{B}(t)$*. Define the factor algebra* $\mathcal{A} := \mathcal{GT}(\mathrm{sig})/\sim$ *. Now:*

1. $\mathcal{A}$ *is a* sig/cons*-model of* R*.*
2. $\mathcal{A} \lesssim_{\mathrm{H}} \mathcal{B}$ *. Moreover, there is a unique (injective)* sig*-homomorphism from* $\mathcal{A}$ *to* $\mathcal{B}$*.*
3. $\mathcal{A} \approx_{\mathrm{cons}} \mathcal{B}$

The following lemma[42] of theoretical nature ensures the existence of minimal models:

Lemma 4.5 *Let* R *be a CRS over* sig/cons/V*.*

1. *The trivial* sig*-algebra is a* sig/cons*-model of* R*.*
2. *If* $\mathcal{B}$ *is a* sig/cons*-model of* R*, then there is a (*$<_{\mathrm{H}}$*-)minimal* sig/cons*-model* $\mathcal{A}$ *of* R *with* $\mathcal{A} \leq_{\mathrm{H}} \mathcal{B}$*.*
3. R *has a minimal* sig/cons*-model.*

---

[40] Cf. Theorem 5.14
[41] Cf. footnote 39
[42] The lemma resembles Theorem 2.1 in Kaplan[23]. Please notice, however, that our $\lesssim_{\mathrm{H}}$ and $\lesssim_{\mathrm{cons}}$ are reflexive and therefore different from the relation $\leq$ in [23], where the homomorphism is additionally required to be unique.

5 The Reduction Relation

In this section we are going to define a reduction relation $\Longrightarrow_{\mathrm{R}}$ which is convenient for the semantics defined in the previous section. The overall idea is to reduce a left-hand side of a rule to its right-hand side only if the condition of this rule can somehow be shown valid by the use of the same reduction relation again.

Many authors impose rather strong restrictions on constructor equations, such as "no equations between constructors" ("free constructors") or "unconditional equations between constructors only". Compared to these, our restrictions are very weak. They serve to guarantee a constructor-minimum model for the constructor equations that is unique modulo $\approx_{\mathrm{cons}}$, by requiring the constructor equations to be "constructor-preserving"[43] and to have "Horn"-form.

Definition 5.1 (Set of Rules)
(continuing Definition 4.1 by adding the restrictions on constructor equations)
The set of rules *over* sig, cons *and* V *is defined to be:*

$$\mathcal{RUL}(\mathrm{sig},\mathrm{cons}) \;:=\; \left\{ \begin{array}{l} ((l,r),C) \in (\mathrm{DEq}(\mathrm{sig}) \times (\mathcal{LIT}(\mathrm{sig}))^*) \\ \left| \;\left(l \in \mathcal{T}(\mathrm{cons}) \;\Rightarrow\; \left(\begin{array}{lr} \mathcal{V}(r, \mathcal{TERMS}(C)) \;\subseteq\; \mathcal{V}(l) & \wedge \\ r \in \mathcal{T}(\mathrm{cons}) & \wedge \\ \mathcal{TERMS}(C) \;\subseteq\; \mathcal{T}(\mathrm{cons}) & \wedge \\ \forall L \textit{ in } C : \forall u,v :\;\; L \neq (u{\neq}v) & \end{array} \right) \right) \right. \end{array} \right\}$$

We are now going to define our reduction relation, having in mind to require it to be confluent in the sequel, whereas we do not require confluence for the definition because we cannot prove confluence criteria if the non-confluent case is undefined. Therefore, we have to be explicit how we test the condition literals — even if this testing is not straightforward when confluence is not provided. Our "operational" semantics for testing condition literals is the following: '$u{=}v$' is true if u, v have reducts $\hat{u}, \hat{v}$, resp., which are syntactically equal. '$\mathrm{Def}\,u$' is true if u has a constructor ground reduct. '$u \neq v$' is true if u, v have constructor ground reducts $\hat{u}, \hat{v}$, resp., which are not joinable. Thus, two terms in a condition literal are "operationally" equal if they are joinable, whereas they are unequal if they are not joinable after some reduction to constructor ground terms. The non-joinability alone of two terms is not sufficient for regarding them as unequal because we are never sure about the inequality of "undefined" terms. As it often occurs, our operational logic is four-valued (i. e. $=$ and $\neq$ can independently be true or false), but in case of confluence: *tertium non datur.* In case of free or confluent constructors, the case of both '$u{=}v$' and '$u{\neq}v$' being true simultaneously means that we have something like an ambiguous function definition.

[43] The constructor-preservation is really necessary here for guaranteeing a minimal constructor-minimum model as in Theorem 5.14: Let $\mathsf{0}, \mathsf{1}, \mathsf{true}, \mathsf{false}$ be constructor constants, let weirdp be a defined constant, and take R:

$$\begin{array}{lcll} \mathsf{1} & = \mathsf{0} & \longleftarrow & \mathsf{weirdp} = \mathsf{true} \quad ; \\ \mathsf{weirdp} & = \mathsf{true} & \longleftarrow & \mathsf{true} \neq \mathsf{false} \quad . \end{array}$$

Now there are models of R with $\mathsf{0} \neq \mathsf{1}$ and models with $\mathsf{true} \neq \mathsf{false}$ but no models with $\mathsf{0} \neq \mathsf{1}$ and $\mathsf{true} \neq \mathsf{false}$. Also notice, that the constructor-preservation has a lot of additional advantages, e. g.:

1. For $u \in \mathcal{GT}(\mathrm{sig})$ and computable and unique normal form $\mathrm{NF}(u)$ we can test
   " $\exists \hat{u} \in \mathcal{GT}(\mathrm{cons}) : u \overset{\circledast}{\Longrightarrow} \hat{u}$ " by " $\mathrm{NF}(u) \in \mathcal{GT}(\mathrm{cons})$ ".
2. Theorem 5.15 has no reasonable analogue for CRSs which are not constructor-preserving.

Another important aspect of our reduction relation is that it depends on the constructor subsignature 'cons' beyond the signature 'sig' — just as our notion of a "cons/sig-model" does.

Definition 5.2 (Our Reduction Relation $\Longrightarrow_{\mathrm{R}}$)
Let R *be a CRS over* sig/cons/V. *The reduction relation* $\Longrightarrow_{\mathrm{R}}$ *on* $\mathcal{T}(\mathrm{sig})$ *($\Longrightarrow$ for short) is defined to be the smallest relation satisfying the following requirement (:#):*

$$s\Longrightarrow t \ \ \underline{\mathit{if}} \ \ \exists((l,r),C)\in\mathrm{R}\colon \exists\sigma\in\mathcal{SUB}(\mathrm{V},\mathcal{T}(\mathrm{sig}))\colon \exists p\in\mathcal{O}(s)\colon \begin{pmatrix} s/p \ = \ l\sigma & \wedge \\ t \ \ = \ s[\,p\leftarrow r\sigma\,] & \wedge \\ C\sigma \ \textit{is}\ \textbf{fulfilled w. r. t.} \Longrightarrow \end{pmatrix},$$

where "C is **fulfilled w. r. t.** $\Longrightarrow$*" is a shorthand for*

$$\forall u,v\in\mathcal{T}(\mathrm{sig})\colon \begin{pmatrix} (\ (\ (u{=}v\ )\ \mathit{in}\ C\ ) & \Rightarrow & & u\downarrow v\ \ ) & \wedge \\ (\ (\ (\mathrm{Def}\,u)\ \mathit{in}\ C\ ) & \Rightarrow & \exists\hat{u}\in\mathcal{GT}(\mathrm{cons}): & u\overset{\circledast}{\Longrightarrow}\hat{u}\ ) & \wedge \\ (\ (\ (u{\neq}v\ )\ \mathit{in}\ C\ ) & \Rightarrow & \exists\hat{u},\hat{v}\in\mathcal{GT}(\mathrm{cons}): & u\overset{\circledast}{\Longrightarrow}\hat{u}\not\downarrow\hat{v}\overset{\circledast}{\Longleftarrow}v\ )^{44} & \end{pmatrix}$$

Usually one tries to find a minimal reduction relation by taking the closure over a finitary generating relation. This is not possible here, because we have a negative condition ($\not\downarrow$). By the "Horn"-form of our constructor equations (and the constructor-preservation), however, we know that this negative condition does not influence the reduction of constructor terms; and (in the definition) '$\not\downarrow$' is applied to constructor (ground) terms only. Thus, we can get our minimal reduction relation by a double closure: first for constructor terms only; second for general terms knowing the constructor reduction to remain unchanged. This two step construction does not destroy the uniformity of the defining requirement(#), which allows us to write uniform normal form procedures, provided that the normal forms are computable.

Define $\Longrightarrow_{\mathrm{R},0} := \emptyset$ and $\Longrightarrow_{\mathrm{R},i+1}$ to be the left-hand side[45] of the requirement(#) of Definition 5.2 with $\Longrightarrow_{\mathrm{R},i}$ substituted for $\Longrightarrow$ on the right-hand side and the additional restriction of $s\in\mathcal{T}(\mathrm{cons})$; formally: $s\Longrightarrow_{\mathrm{R},i+1}t$ $\ \underline{:\mathrm{iff}}\ $ $s\in\mathcal{T}(\mathrm{cons})\wedge$

$$\exists((l,r),C)\in\mathrm{R}\colon \exists\sigma\in\mathcal{SUB}(\mathrm{V},\mathcal{T}(\mathrm{sig}))\colon \exists p\in\mathcal{O}(s)\colon \begin{pmatrix} s/p \ = \ l\sigma & \wedge \\ t \ \ = \ s[\,p\leftarrow r\sigma\,] & \wedge \\ C\sigma \ \text{is}\ \textbf{fulfilled w. r. t.} \Longrightarrow_{\mathrm{R},i} \end{pmatrix}.$$

[44]This formulation requires confluence and constructor-preservation to make sense in two-valued logic. While other formulations (e. g. a universal instead of the existential quantification) might seem more satisfactory, this is the one required for a correct definition.

One might have expected $u\not\downarrow v$ instead of $\exists\hat{u},\hat{v}\in\mathcal{GT}(\mathrm{cons}): u\overset{\circledast}{\Longrightarrow}\hat{u}\not\downarrow\hat{v}\overset{\circledast}{\Longleftarrow}v$ for "$u{\neq}v$ in C" here, but this modification would not allow us to conclude that $\Longrightarrow_{\mathrm{R}}$ is uniquely defined as can be seen from the following example:

Example 5.3
Let $\mathsf{a},\mathsf{b},\mathsf{c},\mathsf{d},\mathsf{e}$ *be constants and take* R*:*

$$\begin{array}{ll} \mathsf{a}=\mathsf{c}\longleftarrow\mathsf{b}\neq\mathsf{d} & ; \\ \mathsf{b}=\mathsf{d}\longleftarrow\mathsf{e}\neq\mathsf{c} & ; \\ \mathsf{e}=\mathsf{a} & . \end{array}$$

The intersection of $\{\ (\mathsf{a},\mathsf{c}),\ (\mathsf{e},\mathsf{a})\ \}$ *and* $\{\ (\mathsf{b},\mathsf{d}),\ (\mathsf{e},\mathsf{a})\ \}$ *(which would be minimal relations satisfying the modified requirement(#) of Definition 5.2) would not satisfy the modified requirement of Definition 5.2 anymore.*

[45]the ' $\underline{\mathrm{if}}$ ' replaced by ' $\underline{:\mathrm{iff}}$ '

Define $\Longrightarrow_{\mathrm{R},\omega} := \bigcup_{i\in\mathbb{N}} \Longrightarrow_{\mathrm{R},i}$ and $\Longrightarrow_{\mathrm{R},\omega+i+1}$ to be the union of $\Longrightarrow_{\mathrm{R},\omega}$ and the left-hand side[45] of the requirement(#) of Definition 5.2 with $\Longrightarrow_{\mathrm{R},\omega+i}$ substituted for $\Longrightarrow$ on the right-hand side. Finally, define $\Longrightarrow_{\mathrm{R}} := \Longrightarrow_{\mathrm{R},\omega+\omega} := \bigcup_{i\in\mathbb{N}} \Longrightarrow_{\mathrm{R},\omega+i}$. Now $\Longrightarrow_{\mathrm{R}}$ satisfies the requirement(#) of Definition 5.2, and every relation satisfying this requirement must contain $\Longrightarrow_{\mathrm{R}}$. Hence $\Longrightarrow_{\mathrm{R}}$ is the intended smallest relation. A more detailed proof for all this can be found in the appendix. By double induction over the above construction process it is trival to verify the following corollaries.

Corollary 5.4 (Monotony of $\Longrightarrow_{\mathrm{R}}$ w. r. t. Replacement)
$\Longrightarrow_{\mathrm{R},i}$, $\Longrightarrow_{\mathrm{R},\omega+i}$ *(for $i\in\mathbb{N}$), $\Longrightarrow_{\mathrm{R}}$ are monotonic.*

Corollary 5.5 (Stability of $\Longrightarrow_{\mathrm{R}}$)
$\Longrightarrow_{\mathrm{R},i}$, $\Longrightarrow_{\mathrm{R},\omega+i}$ *(for $i\in\mathbb{N}$), $\Longrightarrow_{\mathrm{R}}$ and their respective fulfilledness-predicates are stable.*

Lemma 5.6 (Monotony of $\Longrightarrow_{\mathrm{R},\beta}$ in β) *For two ordinal numbers β,γ smaller than $\omega+\omega$, if β is smaller than γ (i. e. if $\beta \preceq \gamma$), then $\Longrightarrow_{\mathrm{R},\beta} \subseteq \Longrightarrow_{\mathrm{R},\gamma}$.*

The following technical lemmas state constructor-preservation and that there is no need for a second closure for reduction of constructor terms.

Lemma 5.7 $\forall n \in \mathbb{N} : \forall s \in \mathcal{T}(\mathrm{cons}) : \forall t : \Big(\ s \overset{n}{\Longrightarrow} t \ \ \Rightarrow \ \ (s \overset{n}{\Longrightarrow}_{\mathrm{R},\omega} t \in \mathcal{T}(\mathrm{cons})) \ \Big)$

Lemma 5.8 $\forall n \in \mathbb{N} : \forall s \in \mathcal{GT}(\mathrm{cons}) : \forall t : \Big(\ s \overset{n}{\Longrightarrow} t \ \ \Rightarrow \ \ (s \overset{n}{\Longrightarrow}_{\mathrm{R},\omega} t \in \mathcal{GT}(\mathrm{cons})) \ \Big)$

Lemma 5.9 $\downarrow \cap (\mathcal{T}(\mathrm{cons}) \times \mathcal{T}(\mathrm{cons})) \ = \ \downarrow_{\mathrm{R},\omega}$

Lemma 5.10 (Fulfilledness Test may be Simple)
Let $C \in \mathcal{LIT}(\mathrm{sig})^*$. If for each element $u \in \mathcal{TERMS}(C)$:

1. *u has a [ground] normal form $\mathrm{NF}(u)$*
   (i. e. $u \overset{\circledast}{\Longrightarrow} \mathrm{NF}(u) \notin \mathrm{dom}(\Longrightarrow)$[, $\mathrm{NF}(u) \in \mathcal{GT}(\mathrm{sig})$]) and
2. *$\Longrightarrow$ is [ground] confluent below u [and $\mathcal{V}(\mathcal{TERMS}(C)) = \emptyset$].*

then C is fulfilled w. r. t. $\Longrightarrow$ <u>iff</u>

$$\forall u,v \in \mathcal{T}(\mathrm{sig}): \begin{pmatrix} (\ (\ (u{=}v \ ) \ \textit{in } C) & \Rightarrow & \mathrm{NF}(u) = \mathrm{NF}(v) \ \ ) & \wedge \\ (\ (\ (\mathrm{Def}\, u) \ \textit{in } C) & \Rightarrow & \mathrm{NF}(u) \in \mathcal{GT}(\mathrm{cons}) \ ) & \wedge \\ (\ (\ (u{\neq}v \ ) \ \textit{in } C) & \Rightarrow & (\mathrm{NF}(u), \mathrm{NF}(v) \in \mathcal{GT}(\mathrm{cons}) \ \wedge \ \mathrm{NF}(u) {\neq} \mathrm{NF}(v)) \ ) & \end{pmatrix}$$

Considering ground confluence, it is important to realize that for reduction on ground terms we can restrict ourselves to ground substitutions:

Lemma 5.11 (Ground Substitutions are Sufficient for Ground Reduction)
Let $\Longrightarrow_{\mathrm{R,ground}}$ be defined just like $\Longrightarrow_{\mathrm{R}}$ with the exception of the additional restriction "$\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig}))$" in the requirement(#) of Definition 5.2.
Now $\Longrightarrow_{\mathrm{R,ground}}$ is uniquely defined, $\Longrightarrow_{\mathrm{R,ground}} = \Longrightarrow_{\mathrm{R}} \cap (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))$,
$\overset{\oplus}{\Longrightarrow}_{\mathrm{R,ground}} = \overset{\oplus}{\Longrightarrow}_{\mathrm{R}} \cap (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))$, *and* $\mathrm{dom}(\Longrightarrow_{\mathrm{R,ground}}) = \mathrm{dom}(\Longrightarrow_{\mathrm{R}}) \cap \mathcal{GT}(\mathrm{sig})$.

By Example 5.3 on page 20 we know that (for guaranteeing a constructor-minimum model) we have to restrict the terms of the negated equations to be "defined". This semantic restriction is made syntactically explicit in the following definition that specifies a "well-behaved" subclass of the class of CRSs, in which inequalities are founded on constructor ground terms.

Definition 5.12 (Moderate Conditional Rule Systems (MCRS))
A CRS R *is a* moderate conditional rule system *(*MCRS*)* *:iff*

$$\forall((l,r),C) \in \mathrm{R} : \forall(u{\neq}v) \textit{ in } C : \Big(\ \mathrm{Def}\,u, \mathrm{Def}\,v \textit{ are in } C\ \Big)\ .$$

For a motivation of MCRSs see item 1 on page 14, where we discussed the problems involved.

The following definition allows us to use a CRS as a shorthand for a MCRS:

Definition 5.13 (Moderated Version of a CRS)
The moderated version of a CRS R *is the MCRS* R′ *obtained by replacing each sequence of the form '*$u{\neq}v$*' in each condition of each rule in* R *by the*[46] *sequence '*$u{\neq}v,\ \mathrm{Def}\,u,\ \mathrm{Def}\,v$*'.*

Now we are able to state the fundamental theorem about $\Longrightarrow$. The theorem states that for moderate CRS R with confluent $\Longrightarrow_{\mathrm{R}}$ the factor algebra $\mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}_{\mathrm{R}}$ is an (up to isomorphism) uniquely determined cons/sig-model of R.

Theorem 5.14
(Minimal Model being the Minimum of the Constructor-Minimal Models)

Let R *be a MCRS over* sig/cons/V*.*

Furthermore, assume $\Longrightarrow_{\mathrm{R}}$ *to be confluent.*

1. $\mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}$ *is a constructor-minimum*[47] *model of* R*.*

2. *Let* $\mathcal{B}$ *be a model of* R *with*

   (a) $\mathcal{B}$ *a constructor-minimal*[47] *model of* R
   or
   (b) $\mathcal{B} \lesssim_{\mathrm{H}} \mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}$ *.*

   Now $\mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow} \lesssim_{\mathrm{H}} \mathcal{B}$*.*

   In other words: $\mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}$ *is (by 2b) a minimal model and (by 2a) the minimum of all models that do not identify more constructor ground terms than necessary.*

[46] We think the symmetry in u and v to be removed in some deterministic manner
[47] Cf. Definition 4.3 on page 18

Theorem 5.15
(Monotony of $\Longrightarrow_{\mathrm{R}}$ w. r. t. Consistent Extension of the Specification)

Let R *be a CRS over* sig, cons *and* V*. Let* R' *be another CRS, but over* sig', cons' *and* V'*;* *with*

$$\left|\begin{array}{rl} \mathrm{sig}' &= (\mathrm{F}',\mathrm{S}',\alpha') \\ \mathrm{cons}' &= (\mathrm{C}',\mathrm{S}',\alpha'|_{\mathrm{C}'}) \\ \mathrm{V}'|_{\mathrm{S}} &= \mathrm{V} \\ & \end{array}\right|\begin{array}{l} \mathrm{F} \subseteq \mathrm{F}' \\ \mathrm{C} \subseteq \mathrm{C}' \subseteq \mathrm{F}' \\ \mathrm{S} \subseteq \mathrm{S}' \\ \alpha \subseteq \alpha' \end{array}\left|\begin{array}{l} \mathrm{R} \subseteq \mathrm{R}' \end{array}\right|$$

Thus, $\mathrm{sig}'/\mathrm{cons}'/\mathrm{V}'$ *is an enrichment of* $\mathrm{sig}/\mathrm{cons}/\mathrm{V}$ *in the most general*[48] *sense we can think of.*

Moreover, assume[49]*:* $\forall((l,r),C) \in (\mathrm{R}' \setminus \mathrm{R}) : l \notin \mathcal{T}(\mathrm{cons})$ *(:*$\$$*)*

Now we have:

1. $\forall s \in \mathcal{T}(\mathrm{cons}) \colon \forall t \colon \Big((s \stackrel{\circledast}{\Longrightarrow}_{\mathrm{R}} t) \;\Leftrightarrow\; (s \stackrel{\circledast}{\Longrightarrow}_{\mathrm{R}'} t) \Big)$ *"no change on old constructor terms"*
2. $\Longrightarrow_{\mathrm{R}} \;\subseteq\; \Longrightarrow_{\mathrm{R}'}$ *"monotony"*

---

[48] One may even define new constructor function symbols for the old sorts and take them from the old defined symbols.

[49] This has to be required for keeping the negative conditions fulfilled: Having founded our inequalities on old constructor ground terms, all we have to take care of now is not to confuse these terms.

6 How to Test for Confluence

We are now going to define critical peaks that consist of the conditional critical pair, its peak, and the overlap position. The other notions we will use are standard, with the exception of the notions "weakly joinable" and "quasi overlay joinable" which are slight weakenings of "joinable" and "overlay joinable", resp., in Dershowitz[12] (cf. below). The first one adds confluence below the condition terms to the premise; the second one allows a constant non-overlay part in the critical pair.

Definition 6.1 (Critical Peaks and Joinability)
The set of (non-trival) critical peaks *between two rules* $((l_k, r_k), C_k) \in \mathrm{R}$ *;* $k < 2$ *; is defined as:*

$$
\begin{array}{ll}
\multicolumn{2}{l}{\mathrm{Cp}(\ ((l_0, r_0), C_0),} \\
\multicolumn{2}{l}{\qquad ((l_1, r_1), C_1)\ )} \\
:= & \\
\{ & (((l_1[\,p \leftarrow r_0\xi\,], r_1), C_0\xi\, C_1)\sigma,\quad l_1\sigma,\quad p) \\
| & \xi = \min \mathrm{Sep}(\mathcal{V}(((l_0, r_0), C_0)), \mathcal{V}(((l_1, r_1), C_1))) \ \wedge \qquad \text{“no variables in common”} \\
& p \in \mathcal{O}(l_1)\ \wedge\ l_1/p \notin \mathrm{V} \ \wedge \qquad \text{“non-variable position”} \\
& \sigma = \min \mathrm{Mgu}(l_0\xi, l_1/p) \ \wedge \qquad \text{“most general unifier”} \\
& l_1[\,p \leftarrow r_0\xi\,]\sigma \neq r_1\sigma \qquad \text{“non-trivial critical pair”} \\
\} &
\end{array}
$$

And the set of all critical peaks of R *is* $\mathrm{CP(R)} := \bigcup_{\mathrm{rule}_0 \in \mathrm{R}} \bigcup_{\mathrm{rule}_1 \in \mathrm{R}} \mathrm{Cp}(\mathrm{rule}_0, \mathrm{rule}_1)$.
R *is said to be* overlapping <u>*:iff*</u> $\mathrm{CP(R)} \neq \emptyset$.
A critical peak $(((t_0, t_1), D), \hat{t}, p)$ *is* weakly [ground] joinable *(w. r. t.* $\Longrightarrow$*)* <u>*:iff*</u>

$$\forall \varphi \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})]) : \left(\begin{pmatrix} D\varphi \textit{ fulfilled} & \wedge \\ \forall u \in \mathcal{TERMS}(D\varphi): & \\ \quad \Longrightarrow \textit{ is [ground] confluent below } u & \end{pmatrix} \Rightarrow\ t_0\varphi \downarrow t_1\varphi \right).$$

A critical peak $(((t_0, t_1), D), \hat{t}, p)$ *is* [ground] joinable *(w. r. t.* $\Longrightarrow$*)* <u>*:iff*</u>

$$\forall \varphi \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})]) : ((D\varphi \textit{ fulfilled})\ \Rightarrow\ t_0\varphi \downarrow t_1\varphi)\ .$$

A critical peak $(((t_0, t_1), D), \hat{t}, p)$ *is* quasi overlay [ground] joinable *(w. r. t.* $\Longrightarrow$*)* <u>*:iff*</u>

$$\forall \varphi \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})]) \text{:} \left((D\varphi \textit{ fulfilled})\ \Rightarrow \begin{pmatrix} t_1\varphi = t_0\varphi[\,p \leftarrow t_1\varphi/p\,] & \wedge \\ (t_0/p)\varphi \downarrow t_1\varphi/p \overset{\oplus}{\Longleftarrow} (\hat{t}/p)\varphi & \end{pmatrix} \right).$$

A critical peak $(((t_0, t_1), D), \hat{t}, p)$ *is* overlay [ground] joinable *(w. r. t.* $\Longrightarrow$*)* <u>*:iff*</u> *it is [ground] joinable and* $p = \emptyset$*.*

Lemma 6.2 (Joinability of Critical Peaks is Necessary for Confluence)
Let R *be a CRS over* sig, cons *and* V*. If* $\Longrightarrow_{\mathrm{R}}$ *is [ground] confluent, then all critial peaks in* CP(R) *are [ground] joinable (w. r. t.* $\Longrightarrow_{\mathrm{R}}$*).*

Lemma 6.3 (Overlay Joinable $\Rightarrow$ Quasi Overlay Joinable $\Rightarrow$ Joinable)
Let R *be a CRS over* sig/cons/V *and* $(((t_0, t_1), D), \hat{t}, p) \in \mathrm{CP(R)}$*. Now w. r. t.* $\Longrightarrow_{\mathrm{R}}$ *the following holds:*

1. *If* $(((t_0, t_1), D), \hat{t}, p)$ *is overlay [ground] joinable, then it is quasi overlay [ground] joinable.*
2. *If* $(((t_0, t_1), D), \hat{t}, p)$ *is quasi overlay [ground] joinable, then it is [ground] joinable.*

Sufficient criteria for confluence of reduction relations for merely positive conditional rule systems are studied in Dershowitz[12]. As counterexamples for suggested sufficient confluence criteria for merely positive conditional rule systems are counterexamples for MCRS too, we repeat the results of [12] here: There are (left-linear) non-overlapping positive-conditional rule systems whose reduction relations are not (locally) confluent.[50] These rule systems are necessarily non-noetherian.[51] Therefore, syntactic confluence criteria for non-noetherian conditional rule systems must be very difficult to develop. Semantic confluence criteria (in the style of Plaisted[31]) seem to require noetherian (or at least normalising) reduction relations because they rely on the irreducible reducts of the terms; furthermore irreducibility is not (semi-) decidable. Thus, for our confluence criteria we require $\Longrightarrow_{\mathrm{R}}$ to be noetherian . Even then the situation is not very encouraging, because there are noetherian and non-confluent reduction relations of (left-linear, normal, and) positive-conditional rule systems with joinable critical peaks only.[52] Moreover, semantic confluence criteria remain difficult because irreducibility is still not (semi-) decidable. However, for merely positive conditional rule systems there are two known solutions of major[53] interest: One requires either more than joinability for the critical peaks (as, e. g., in Theorem 4 in [12]) or the condition terms to be somehow smaller than the left-hand side of the rule (as, e. g., in Theorem 3 in [12]). We will study the latter approach later[56]. The following result is a generalization of Theorem 4 in Dershowitz[12] from positive-conditional to positive/negative-conditional rule systems and, moreover, from overlay joinability to quasi overlay joinability.

Theorem 6.4 (Confluence Criterion)
Let R *be a CRS over* sig, cons *and* V. *If* $\Longrightarrow_{\mathrm{R}}$ *is [ground] noetherian and all critical peaks in* CP(R) *are quasi overlay [ground] joinable, then* $\Longrightarrow_{\mathrm{R}}$ *is [ground] confluent.*

While this theorem is very nice (theoretically) and has a pretty complicated proof, it may be difficult to apply even for merely positive conditional equations:

Example 6.5 *Let* R*:*
$$\begin{array}{l} \mathsf{f}\,\mathsf{s}\,x{=}\mathsf{0} \longleftarrow \mathsf{f}\,x{=}\mathsf{0} \\ \mathsf{f}\,\mathsf{s}\,x{=}\mathsf{1} \longleftarrow \mathsf{f}\,x{=}\mathsf{1} \\ \mathsf{f}\ldots{=} \end{array}$$
Assume $\mathsf{0}$ *and* $\mathsf{1}$ *to be irreducible. Now for showing the critical peak between the first two rules to be quasi overlay joinable, one has to show that it is impossible that both conditions hold simulaneously for a substitution* $\{x\mapsto t\}$. *However, to know this we need the confluence below '*$\mathsf{f}\,t$*', which we are not allowed to assume for the joinability test here.*

For a confluence criterion of practial use see Theorem 7.14.

[50] Cf. [12], Example A, p. 36
[51] Cf. [12], Theorem 4, p. 39
[52] Cf. [12], Example B, p. 36
[53] We are not interested in the shallow-joinability of Dershowitz[12] here because there is a noetherian, shallow-joinable, left-linear, but not confluent, merely positive conditional rule system[54] as well as a noetherian, shallow-joinable, normal, but not confluent, merely positive conditional rule system[55], which means that shallow-joinability is only sufficient for confluence of left-linear and normal rule systems. The combination of left-linearity and normality, however, is a too severe restriction to be of interest for us, because it means logic without equality.
[54] Cf. [12], Example C, p. 36
[55] Cf. [12], Example D, p. 36
[56] Cf. Theorem 7.14

7 Decidability of the Reduction Relation

In this section we overcome one serious problem of the reduction relation of the previous section: Reducibility of a given term is neither semi- nor co-semi-decidable. Our means of tackling this problem are restrictions on variable occurrence, and compatibility restrictions on the rule systems w. r. t. well-founded orderings.

7.1 Undecidability

Lemma 7.1 (Ground Reducibility is not Co-Semi-Decidable)
There is a non-overlapping, extra-variable free, merely positive conditional rule system R *with noetherian and confluent reduction relation* $\Longrightarrow_{\mathrm{R}}$ *for which reducibility of ground terms is not co-semi-decidable.*

Lemma 7.2 (Ground Reducibility is not Semi-Decidable)
There is a non-overlapping, extra-variable free MCRS R *with noetherian and confluent reduction relation* $\Longrightarrow_{\mathrm{R}}$ *for which reducibility of ground terms is not semi-decidable.*

The following lemma is a generalization of Theorem 3.4 in Kaplan[22] in various directions.

Lemma 7.3
Let R *be a CRS over* sig/cons/V*.*

1. $\Longrightarrow_{\mathrm{R}}$*[-ground]-reducibility is co-semi-decidable if a [ground] normal form for each [ground] term is computable (i. e. there is a computable (partial) function* f *with* $\mathrm{dom}(f) = \{s \in \mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})] \mid \exists t \in \mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})] : (s \overset{\circledast}{\Longrightarrow}_{\mathrm{R}} t \wedge t \notin \mathrm{dom}(\Longrightarrow_{\mathrm{R}}))\}$ *such that* $\forall s \in \mathrm{dom}(f) : s \overset{\circledast}{\Longrightarrow}_{\mathrm{R}} f(s) \in (\mathcal{T}(\mathrm{sig})[\cap \mathcal{GT}(\mathrm{sig})]) \setminus \mathrm{dom}(\Longrightarrow_{\mathrm{R}})$*).*

2. *A [ground] normal form for each [ground] term is computable (see above) if* $\Longrightarrow_{\mathrm{R}}$*[-ground]-reducibility is co-semi-decidable and*
$\forall s \in \mathcal{GT}(\mathrm{cons}) : \exists t : s \overset{\circledast}{\Longrightarrow}_{\mathrm{R}} t \notin \mathrm{dom}(\Longrightarrow_{\mathrm{R}})$.

Corollary 7.4
Let R *be a CRS over* sig/cons/V*. Assume* $\Longrightarrow_{\mathrm{R}}$ *to be noetherian. Now, co-semi-decidability of* $\Longrightarrow_{\mathrm{R}}$*[-ground]-reducibility is logically equivalent to computability of a [ground] normal form for each [ground] term.*

7.2 The Use of Orderings

In this and the following sections our view on orderings and ordering properties is not that of the ordering specialist who is interested in the development of orderings and the effective testing of ordering properties; we just want to give minimal reasonable requirements for achieving additional properties of our reduction relation. While the problem of operationalization is simple for some important special cases, it is difficult in general and beyond the scope of this paper.

We start this section with a general discussion how to use orderings for reduction with conditional rules. This discussion is somewhat independent from the concrete form of our rules and mainly depends on the technique of testing the conditions uniformly by the same reduction relation again, where well-founded orderings are needed for guaranteeing termination of condition-testing and reduction. By this independence, the situation under discussion does not differ from the case of merely positive conditional equations. The reader not interested in this subject may skip it and continue reading with Definition 7.7.

Let $>$ be a well-founded ordering intended to contain the reduction relation $\Longrightarrow$. We require it to be sort-invariant, monotonic, and stable. This requirement does not restrict applicability because the transitive closure $\overset{\oplus}{\Longrightarrow}$ of $\Longrightarrow$ is a sort-invariant, monotonic, and stable ordering itself, provided that it is well-founded.

Let $\hookrightarrow$ be the relation intended to contain the step from a left-hand side's superterm to the condition terms. As a condition term that can be tested for the attempt to reduce l should also be testable for the reduction of $t[\,p \leftarrow l\,]$ (otherwise $\overset{\oplus}{\Longrightarrow}$ wouldn't be monotonic), we require the property $\rhd_{\mathrm{ST}} \circ \hookrightarrow \subseteq \hookrightarrow$, — considering $t[\,p \leftarrow l\,] \trianglerighteq_{\mathrm{ST}} l \hookrightarrow u \;\Rightarrow\; t[\,p \leftarrow l\,] \hookrightarrow u$.

As we test our conditions by reduction we must be allowed to switch from reduction to condition-testing, and then to reduction of the condition terms, and so on. Hence we need the transitive closure of $>$ and $\hookrightarrow$ to be noetherian. Thus, in the sequel we require $\rightsquigarrow \;:= (> \cup \hookrightarrow)^{\oplus}$ to be noetherian.

For going on we need the following two lemmas:

Lemma 7.5 *Let $>$ be a monotonic ordering on $\mathcal{T}(\mathrm{sig})$. Define* $\succ := \ \rhd_{\mathrm{ST}} \cup (> \circ \trianglerighteq_{\mathrm{ST}})$. *Now the following holds:*

1. $\rhd_{\mathrm{ST}} \circ > \ \subseteq \ > \circ \rhd_{\mathrm{ST}}$
2. $\succ \ = \ (> \cup \rhd_{\mathrm{ST}})^{\oplus}$
3. *If $>$ is sort-invariant*[57] *[,stable,] and well-founded, then $\succ$ is a well-founded [and stable] ordering, which doesn't need to be sort-invariant or monotonic.*
4. *(3) does not hold in general if $>$ is not sort-invariant or not monotonic.*

Lemma 7.6
Let[58] $\rhd_{\mathrm{ST}} \circ > \ \subseteq \ > \circ \rhd_{\mathrm{ST}}$. *Let*[59] $\rhd_{\mathrm{ST}} \circ \hookrightarrow \ \subseteq \ \hookrightarrow$. *Define* $\rightsquigarrow := (> \cup \hookrightarrow)^{\oplus}$ *and assume it to be noetherian.*[60] *Assume*[61] $(> \cup \rhd_{\mathrm{ST}})^{\oplus}$ *to be noetherian. [Assume $\hookrightarrow$ and $>$ to be stable.] Now:*

$\blacktriangleright := (> \cup \rhd_{\mathrm{ST}} \cup \hookrightarrow)^{\oplus}$ *is noetherian [and stable], too.*

By these two lemmas we see the following: Instead of taking (in addition to $>$) two relations $\hookrightarrow$ and $\rightsquigarrow$ with the above requirements[62], we can take a single relation $\blacktriangleright$ (without giving up the required well-foundedness) and require the stronger $\rhd_{\mathrm{ST}} \subseteq \blacktriangleright$ instead of the weaker $\rhd_{\mathrm{ST}} \circ \hookrightarrow \ \subseteq \ \hookrightarrow$. This justifies the following definition:

Definition 7.7 (Termination-Pair)
A termination-pair *over* sig/V *is a pair* $(>, \blacktriangleright)$ *for which the following properties hold:*

1. $>, \blacktriangleright \ \subseteq \ \mathcal{T}(\mathrm{sig}) \times \mathcal{T}(\mathrm{sig})$
2. *$>$ is a monotonic and stable*[63] *ordering*[64].
3. *$\blacktriangleright$ is a stable*[63] *and well-founded ordering.*
4. $> \ \subseteq \ \blacktriangleright$
5. $\rhd_{\mathrm{ST}} \subseteq \blacktriangleright$ [65]

[57] The easiest way to achieve sort-invariance is to identify all sorts
[58] This matches Lemma 7.5(1).
[59] This was required for $\hookrightarrow$ in our discussion above.
[60] This was required in our discussion above.
[61] This matches the conclusion of Lemma 7.5(3), (using Lemma 7.5(2)).
[62] i. e. $>$ sort-invariant, monotonic, well-founded [, and stable] ordering. $\rhd_{\mathrm{ST}} \circ \hookrightarrow \ \subseteq \ \hookrightarrow$. [$\hookrightarrow$ stable.] $\rightsquigarrow := (> \cup \hookrightarrow)^{\oplus}$ noetherian.
[63] Stability is added because it can always be achieved (for $>$ and $\hookrightarrow$; and thereby for $\blacktriangleright$, too) by restriction to ground terms — and a non-stable ordering on non-ground terms is of no use anyway.
[64] As discussed above, sort-invariance can be required here; but it is of no use for us and omitted for convenience. For the benefit from this see Example 7.8(2).
[65] Notice that we only have to show $\rhd_{\mathrm{ST}} \circ \blacktriangleright \ \subseteq \ \blacktriangleright$, because then we know by Lemma 7.5 (applied to the sort-invariant restriction of $>$), and then by Lemma 7.6, that $(\rhd_{\mathrm{ST}} \cup \blacktriangleright)^{\oplus}$ will do the job of $\blacktriangleright$.

Example 7.8 *The standard examples for a termination-pair* $(>, \rhd)$ *are:*

1. $>$ *some sort-invariant reduction ordering;*
   $\rhd := \rhd_{\mathrm{ST}} \cup (> \circ \unrhd_{\mathrm{ST}})$. [66]

2. $>$ *some simplification ordering;*
   $\rhd := >$.

In the field of ordering restrictions for conditional rule systems the notions of "simplifying" and "reductive" are commonly used with diverse meanings. To avoid misunderstandings we use "aligned" for the local restriction on a single rule and "compatible" (with different prefixes) for the restictions that involve the reduction relation of the whole rule system.

Definition 7.9 (Alignment of a Rule w. r. t. a Termination-Pair)
Let $(>, \rhd)$ *be a termination-pair over* sig/V.
A rule $((l, r), C) \in \mathcal{RUL}(\mathrm{sig}, \mathrm{cons})$ *is called* aligned with $(>, \rhd)$ **:iff**

$$l > r \quad \wedge \quad \forall u \in \mathcal{TERMS}(C) : \; l \rhd u$$

Now if we compare the use of orderings in the field of conditional equations (e. g., for the "decreasing"-property in Dershowitz[13, 12]), we find that there usually is a single noetherian ordering containing $\rhd_{\mathrm{ST}}$. This ordering corresponds[67] to our ordering $\rhd$ of a termination-pair $(>, \rhd)$. Our additional $>$, however, is very useful both in practice and for establishing theoretical properties as we will see in the sequel. Furthermore, our previous discussion reveals how to establish the properties required for $\rhd$ and its interference with rules in practice. If we had to restrict to a single ordering we would prefer $>$, because we can get a theoretically appropriate $\rhd$ by $\rhd := \rhd_{\mathrm{ST}} \cup ((> \cap \bigcup_{s \in \mathrm{S}} (\mathcal{T}(\mathrm{sig})_s \times \mathcal{T}(\mathrm{sig})_s)) \circ \unrhd_{\mathrm{ST}})$, but we prefer to avoid this common overspecification, which may not be practically appropriate for alignment of given rules (cf. the rule system on p. 546 in Dershowitz [13]).

[66] Cf. Lemma 7.5
[67] Stability can always be achieved by restriction to ground terms

7.3 Several kinds of Compatibility of a CRS with a Termination-Pair

In this section we firstly define several kinds of compatibility and secondly exhibit some of their benefits.

The following kind of compatibility is a generalization to negative conditions and a slight weakening of "decreasingness" in Dershowitz[12].

Definition 7.10 ([Ground] Compatibility of a CRS with a Termination-Pair)
A CRS R *over* sig/cons/V *is* [ground] compatible *with a termination-pair* $\mathrm{T} = (>, \rhd)$ *over* sig/V **:iff**

$$\forall((l,r),C) \in \mathrm{R} : \forall\tau \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]) : \\ \Big((C\tau \text{ fulfilled w. r. t. } \Longrightarrow_{\mathrm{R}}) \ \Rightarrow\ \Big(\ ((l,r),C)\tau \text{ is aligned with } \mathrm{T}\ \Big)\Big)$$

The [ground] compatibility of a CRS R with a termination-pair guarantees the alignment of a [ground] instantiated rule of R when its condition is fulfilled. But, while this kind of compatibility is convenient for obtaining further theoretical properties of the reduction relation, we have a problem when using this kind of compatibility of R in practice of reduction: The terms in $C\tau$ must be smaller than $l\tau$ only if $C\tau$ is fulfilled; but for easily deciding whether $C\tau$ is fulfilled we need its terms to be smaller than $l\tau$ and the analogous property for the other rules. That this need not be a vicious circle is shown by the following definition, which allows us to test the literals in the condition from left to right, using the old programming trick of a sequential "short-circuiting" AND-operator[68]. Notice that the difference to Definition 7.10 is in the quantified variable i occurring as an index which allows us to step inductively from $(< i)$ to i.

Definition 7.11 ([Ground] Left-Right-Compatibility)
A CRS R *over* sig/cons/V *is* [ground] left-right-compatible *with a termination-pair* $\mathrm{T} = (>, \rhd)$ *over* sig/V **:iff**

$$\forall((l,r),L_0 \ldots L_{n-1}) \in \mathrm{R} : \forall\tau \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]) : \\ \left(\begin{array}{ll} \forall i < n : (& ((L_0 \ldots L_{i-1})\tau \text{ fulfilled w. r. t. } \Longrightarrow_{\mathrm{R}}) \Rightarrow \forall u \in \mathcal{TERMS}(L_i) : l\tau \rhd u\tau \ ) \\ \wedge \qquad (& ((L_0 \ldots L_{n-1})\tau \text{ fulfilled w. r. t. } \Longrightarrow_{\mathrm{R}}) \Rightarrow l\tau > r\tau \ ) \end{array} \right)$$

Definition 7.12 ([Ground] Don't-Care-Compatibility)
A CRS R *over* sig/cons/V *is* [ground] don't-care-compatible *with a termination-pair* $\mathrm{T} = (>, \rhd)$ *over* sig/V **:iff**

$$\forall((l,r),L_0 \ldots L_{n-1}) \in \mathrm{R} : \forall\tau \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]) : \\ \left(\begin{array}{ll} \forall i < n : & \forall u \in \mathcal{TERMS}(L_i) : l\tau \rhd u\tau \\ \wedge \qquad & (((L_0 \ldots L_{n-1})\tau \text{ fulfilled w. r. t. } \Longrightarrow_{\mathrm{R}}) \Rightarrow l\tau > r\tau \ ) \end{array} \right)$$

[68] '&&' (instead of '&') in C. 'AND' in LISP.

Having a left-right-compatible CRS, we don't have to test the instantiated rules for alignment anymore, provided that we test the literals of the instantiated conditions from left to right until one of them fails. Having a don't-care-compatible CRS, we can even test the literals of the instantiated conditions in parallel and don't have to care for the position of these equations in the condition list. The don't-care-compatibility is conceptionally the same as the "decreasingness" in Dershowitz[12].

The "compatibility" of Definition 7.10 (which seems to be the least restrictive one tractable in theory[69]) is intended to be an interface for the generation of logically stronger kinds of "compatibility" that are useful in practice (cf. definitions 7.11 and 7.12), where the don't-care-compatibility of Definition 7.12 seems to be the most important one. For restrictions of $\Longrightarrow_{\mathrm{R}}$, however, even weaker kinds of "compatibility" than the one of Definition 7.10 may be sufficient.

The usefulness of all these kinds of compatibility is shown (w. r. t. Corollary 7.4) by the following lemma.

Lemma 7.13
Let R *be a CRS over* sig/cons/V *and* $\mathrm{T} = (>, \rhd)$ *a termination-pair over* sig/V*. Assume that* R *is [ground] compatible with* T*.*
Now we have $(\Longrightarrow_{\mathrm{R}}[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]) \subseteq >$*, and* $\Longrightarrow_{\mathrm{R}}$ *is noetherian.*
Furthermore, if R *is extra-variable free and if*

1. $\rhd[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]$ *is decidable*

or

2. R *is [ground] left-right-compatible with* T*,*

then $\Longrightarrow_{\mathrm{R}}$*[-ground]-reducibility is decidable and* $\{\ t \mid s \overset{\oplus}{\Longrightarrow}_{\mathrm{R}} t\ \}$ *is a finite computable set for all* $s \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]$.

For compatible CRSs we can now give a complete confluence test à la Knuth-Bendix:

Theorem 7.14 (Confluence Test)
Let R *be a CRS over* sig/cons/V *that is [ground] compatible with a termination-pair* $\mathrm{T} = (>, \rhd)$ *over* sig/V*. The following two are logically equivalent:*

1. $\Longrightarrow$ *is [ground] confluent.*
2. *All critical peaks in* CP(R) *are (weakly) [ground] joinable.*

[69] allowing a complete confluence test and proving theorems

8 Related Work and Discussion

The presented constructor-based approach for positive/negative-conditional equational specifications provides the theoretical background for inductive theorem proving in such specifications as described in Wirth[33]. Various conceivable notions of inductive validity are presented and discussed there. Moreover, an abstract inference system in the style of [1] for proving inductive (constructor) validity of first-order clauses is given. Several refined versions of this inference system incorporating powerful contextual rewriting and simplification rules as well as a marking scheme are developed. In case of a (ground confluent, noetherian and) sufficiently complete base specification the notions of (ordinary) inductive validity and (inductive) constructor validity coincide.

The idea of choosing a constructor-based approach for assigning adequate semantics to positive/negative-conditional equational specifications was heavily inspired by previous work of Kapur&Musser[25, 26] (for the case of unconditional equations only) and Zhang[34] where problems with the usual notion of inductive (initial) validity are treated, in particular concerning its non-monotonic behaviour w. r. t. consistent extensions, and partially defined functions. Under the appropriate syntactical restrictions (mentioned above) concerning the form of positive/negative-conditional rules it turns out that the combination of these ideas with the approach of Kaplan[23] becomes very fruitful and also relevant for practical purposes since many natural specifications involve both conditional equations (rules) with positive and negative conditions and partially defined functions.

The *perfect model* semantics approach of Bachmair&Ganzinger[4], which also includes a completion procedure, generalizes Kaplan's approach [23] by abstracting the control information hidden in the syntactic form of rules into a reduction ordering which must be total on ground terms and which determines the construction process of perfect models. In our analysis, however, reduction orderings used for orienting positive/negative-conditional rule systems do not necessarily have to be total on ground terms. Very recently, another method for inductive theorem proving of first-order clauses has been developed in Ganzinger&Stuber[16] along the lines of [4] by means of a transformational approach: Based on the work of Bachmair&Ganzinger[3, 4], in particular using the abstract notion of redundancy (of clauses and inferences) defined there, the authors develop the viewpoint of inductive theorem proving as verifying redundancy. Its relation to our approach in theory and practice remains to be investigated and will be subject of future investigations. Also related to our approach is the very recent work of Becker[5] who combines the idea of perfect model semantics with parameterization features.

9 Conclusion

We have presented a novel constructor-based approach to positive/negative-conditional equational specifications. By combining different ideas we have been able to define semantics for such specifications which – under some reasonable assumptions concerning the reduction relation – admits a unique model that is the minimum of the constructor-minimal models. The lack of an initial model was one of the main disadvantages of the approach of Kaplan[23]. Our reduction relation yields a logically unique model if confluence is provided. Moreover, it is stable w. r. t. substitutions and monotonic w. r. t. consistent extensions. By extending results from Dershowitz[12, 13] to positive/negative-conditional rule systems and slightly generalizing them we have also been able to provide some interesting confluence criteria for the new reduction relation. Finally, a thorough and precise analysis of termination and decidability issues has led to some useful and slightly weakened "decreasingness" notions for positive/negative-conditional rule systems. The whole approach may be considered to be a firm theoretical basis for (first-order) inductive theorem proving in theories specified by positive/negative-conditional equationals as developed in Wirth[33].

Acknowledgement:
We would like to thank Ulrich Kühler for many valuable discussions and detailed criticisms on earlier versions of this report.

10 Appendix

Proof of Lemma 4.4

1. On $\mathcal{GT}(\mathrm{sig})$ $\mathcal{A}$ has the same kernel as $\mathcal{B}$: The two do not differ in their evaluation of two ground terms of the same sort regarding '='. Therefore, they do not differ in their evaluation of ground terms of the same sort regarding '$\neq$' or of a ground term regarding 'Def', either. Hence the two do not differ in the evaluation of ground rules.

   Now, if $\mathcal{B}$ is a model of R, by the Substitution-Lemma(2.4), it makes true all ground instances of the rules of R, which implies that $\mathcal{A}$ makes true all ground instances of the rules of R, which again implies that $\mathcal{A}$ makes true all the rules of R. This last step can be seen the following way: Let b be any $\mathcal{A}$-valuation of V. Then[70] there is a $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig}))$ such that $\sigma \circ \mathcal{A} = b$. Then by the Substitution-Lemma $\forall t \in \mathcal{T}(\mathrm{sig}) : \mathcal{A}_b(t) = \mathcal{A}(t\sigma)$.

2. This is nothing but an application of the Homomorphism-Theorem.

3. By (*2*).

Proof of Lemma 4.5

1. Trivial.

2. If $\mathcal{B}$ is minimal itself, we are finished. If not, by Lemma 4.4(2) we can w. l. o. g. assume $\mathcal{B}$ to be a factor algebra of $\mathcal{GT}(\mathrm{sig})$ and consider congruences on $\mathcal{GT}(\mathrm{sig})$ instead of algebras. We only have to apply Zorn's Lemma to get a minimal congruence whose factor algebra is a model of R, because this factor algebra will be a minimal model of R. Therefore, we are now going to show that the premise of Zorn's Lemma is satisfied.

   Let $\mathrm{CHAIN} = \{\sim_i \mid i \in I\}$ be a nonempty $\subseteq$-chain of congruence-relations on $\mathcal{GT}(\mathrm{sig})$ with $\forall i \in I : (\mathcal{GT}(\mathrm{sig})/\sim_i$ is a model of R$)$. Define $\approx := \bigcap_{i \in I} \sim_i$. Of course, $\approx$ is a congruence on $\mathcal{GT}(\mathrm{sig})$ that bounds CHAIN below. The only thing left to show is that $\mathcal{GT}(\mathrm{sig})/\approx$ is a model of R.
   Contrariwise there was[71] a $((l,r),C) \in \mathrm{R}$ and a $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig}))$ with
   $\forall L \text{ in } C : \forall u,v : (L = (\langle u,v\rangle, =) \ \Rightarrow\ u\sigma \approx v\sigma)$;
   $\forall L \text{ in } C : \forall u : (L = (\langle u\rangle, \mathrm{Def}) \ \Rightarrow\ \exists \hat{u} \in \mathcal{GT}(\mathrm{cons}) : u\sigma \approx \hat{u})$;
   $\forall L \text{ in } C : \forall u,v : (L = (\langle u,v\rangle, \neq) \ \Rightarrow\ u\sigma \not\approx v\sigma)$; $l\sigma \not\approx r\sigma$.
   Now, by asking for the 'reasons' of the '$\not\approx$', there is a J with $J \subseteq I$;
   $\forall L \text{ in } C : \forall u,v : (L = (\langle u,v\rangle, \neq) \ \Rightarrow\ \exists j_{\langle u,v\rangle} \in J : u\sigma \not\sim_{j_{\langle u,v\rangle}} v\sigma)$;
   $\exists j_{(l,r)} \in J : l\sigma \not\sim_{j_{(l,r)}} r\sigma$; and $|J| \leq |\{L \text{ in } C \mid \exists u,v : L = (\langle u,v\rangle, \neq)\}| + 1$.
   Define $\equiv := \bigcap_{j \in J} \sim_j$. As $\approx \subseteq \equiv$, $\equiv$ yields no model of R for the same 'reason' as $\approx$. But (as J finite) $\{\sim_j \mid j \in J\}$ has a $\subseteq$-minimal[72] element $\sim_{j_0}$ with $j_0 \in J$.
   Now $\equiv = \sim_{j_0}$. Hence $\sim_{j_0}$ yields no model of R, which is a contradiction.

3. By (*1*) and (*2*).

---

[70] by the Axiom of Choice

[71] We tacitly use the Substitution-Lemma(2.4) again, just the way we used it at the end of (1) in the proof of Lemma 4.4.

[72] and minimum

Correctness Proof for Definition 5.2

Let $\Longrightarrow$ be the intersection of all relations satisfying the requirement of Definition 5.2. We claim the following:

1. $\Longrightarrow_{\mathrm{R},\omega}$ is the minimum of all relations satisfying the requirement of Definition 5.2 with the additional restriction of $s \in \mathcal{T}(\mathrm{cons})$.

2. $\forall s \in \mathcal{T}(\mathrm{cons}) : \forall t : (s \overset{\circledast}{\Longrightarrow}_{\mathrm{R},\omega} t \quad \Rightarrow \quad t \in \mathcal{T}(\mathrm{cons}))$

3. $\forall i \in \mathbb{N} : \forall n \in \mathbb{N} : \ (\overset{n}{\Longrightarrow}_{\mathrm{R},\omega+i} \cap (\mathcal{T}(\mathrm{cons}) \times \mathcal{T}(\mathrm{sig}))) \ \subseteq \ \overset{n}{\Longrightarrow}_{\mathrm{R},\omega}$

4. $\forall i \in \mathbb{N} : \ \Longrightarrow_{\mathrm{R},\omega+i} \ \subseteq \ \Longrightarrow_{\mathrm{R},\omega+i+1}$

5. $\Longrightarrow_{\mathrm{R}}$satisfies the requirement of Definition 5.2; i. e. $\Longrightarrow \subseteq \Longrightarrow_{\mathrm{R}}$.

6. $\Longrightarrow_{\mathrm{R},\omega} \ = \ (\Longrightarrow \cap (\mathcal{T}(\mathrm{cons}) \times \mathcal{T}(\mathrm{sig})))$

7. $\forall i \in \mathbb{N} : \ \Longrightarrow_{\mathrm{R},\omega+i} \ \subseteq \ \Longrightarrow$

8. $\Longrightarrow_{\mathrm{R}}$ is the minimum of all relations which satisfy the requirement of Definition 5.2.

To the proofs of these claims:

1. By the restriction on the rules of Definition 5.1, this is just the standard closure over a finitary relation.

2. By the restriction on the rules of Definition 5.1.

3. By induction on i using (1), (2), and the restriction on the rules of Definition 5.1.

4. By induction on i using (2), (3), and the restriction on the rules of Definition 5.1.

5. By (4), taking $\Longrightarrow_{\mathrm{R},\omega+i+1}$ on the left-hand side of the requirement for the maximum i of all $\Longrightarrow_{\mathrm{R},\omega+i}$ occurring on the right-hand side of the requirement.

6. <u>"$\subseteq$":</u> By (1) $\Longrightarrow_{\mathrm{R},\omega}$ is the intersection of a superset of the set whose intersection is $\Longrightarrow$.
   <u>"$\supseteq$":</u> By (3) and (5).

7. <u>$i = 0$:</u> By (6). <u>$i \Rightarrow (i+1)$:</u> By (6) and (2).

8. By (5) and (7).

<u>Q. e. d. (Correctness Proof for Definition 5.2)</u>

Proof of Lemma 5.6
By (4), (8) in the correctness proof above.
Q. e. d. (Lemma 5.6)

Proof of Lemma 5.7
By (2), (3) in the correctness proof for Definition 5.2 above.
Q. e. d. (Lemma 5.7)

Proof of Lemma 5.8
By the restriction on the rules of Definition 5.1 and induction over i one gets $\forall i \in \mathbb{N} : \forall n \in \mathbb{N} : \forall s \in \mathcal{GT}(\mathrm{cons}) : \forall t : (\ s \stackrel{n}{\Longrightarrow}_{\mathrm{R},i} t \quad \Rightarrow \quad t \in \mathcal{GT}(\mathrm{cons})\ )$. By Lemma 5.7 this is sufficient.
Q. e. d. (Lemma 5.8)

Proof of Lemma 5.9
By Lemma 5.7.
Q. e. d. (Lemma 5.9)

Proof of Lemma 5.10
The 'if'-part of the proof is trivial. The 'only if'-part is straightforward as follows: If $u \downarrow v$, then by confluence (below v) $u \stackrel{\circledast}{\Longrightarrow} \mathrm{NF}(v)$ and then by confluence (below u) $\mathrm{NF}(u) = \mathrm{NF}(v)$. If $u \stackrel{\circledast}{\Longrightarrow} \hat{u} \in \mathcal{GT}(\mathrm{cons})$ then by confluence $\hat{u} \stackrel{\circledast}{\Longrightarrow} \mathrm{NF}(u)$ and then by Lemma 5.8 $\mathrm{NF}(u) \in \mathcal{GT}(\mathrm{cons})$. If $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$ and $u \stackrel{\circledast}{\Longrightarrow} \hat{u} \not\downarrow \hat{v} \stackrel{\circledast}{\Longleftarrow} v$ then by confluence $\mathrm{NF}(u) \stackrel{\circledast}{\Longleftarrow} \hat{u} \not\downarrow \hat{v} \stackrel{\circledast}{\Longrightarrow} \mathrm{NF}(v)$; hence $\mathrm{NF}(u) \neq \mathrm{NF}(v)$ and by Lemma 5.8 $\mathrm{NF}(u), \mathrm{NF}(v) \in \mathcal{GT}(\mathrm{cons})$.

For the ground confluence case one additionally applies Corollary 5.5 for an arbitrary ground substitution before using the ground confluence property.
Q. e. d. (Lemma 5.10)

Proof of Lemma 5.11
The unique definedness is shown in a manner analogous with that of $\Longrightarrow_{\mathrm{R}}$. The first two equalities are trivial (double induction using Corollary 5.5). The last equality follows immediately from Corollary 5.5.
Q. e. d. (Lemma 5.11)

Proof of Theorem 5.14

1. Let $\mathcal{A} := \mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}$. For $\mathcal{A}$ being a model of R it is sufficient to note that by confluence of $\Longrightarrow$, the fact that R is a **MCRS**, and Lemma 5.8:
   C is fulfilled <u>iff</u>
   $$\forall L \text{ in } C : \left(\begin{array}{lcll} (L = (\langle u,v\rangle, =\ ) & \Rightarrow & & u \overset{\circledast}{\Longleftrightarrow} v\ ) \ \wedge \\ (L = (\langle u\ \ \rangle, \mathrm{Def}) & \Rightarrow & \exists \hat{u} \in \mathcal{GT}(\mathrm{cons}) : & u \overset{\circledast}{\Longleftrightarrow} \hat{u}\ ) \ \wedge \\ (L = (\langle u,v\rangle, \neq\ ) & \Rightarrow & & u \overset{\circledast}{\not\Longleftrightarrow} v\ ) \end{array}\right)$$
   For the proof of $\mathcal{A}$ to be a constructor-minimum model suppose $\mathcal{B}$ to be a model of R. Define $\sim$ as in Lemma 4.4.
   <u>Claim 1:</u> $\forall i \in \mathbb{N} : (\Longrightarrow_{\mathrm{R},i} \cap\, (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))) \ \subseteq \sim$
   <u>Proof of Claim 1:</u> By induction on i. <u>Q. e. d. (Claim 1)</u>

   Using Lemma 4.4 (3) all we have to show is $(\overset{\circledast}{\Longleftrightarrow} \cap\, (\mathcal{GT}(\mathrm{cons}) \times \mathcal{GT}(\mathrm{cons}))) \ \subseteq \sim$, which by confluence of $\Longrightarrow$ is the same as $(\downarrow \cap (\mathcal{GT}(\mathrm{cons}) \times \mathcal{GT}(\mathrm{cons}))) \ \subseteq \sim$. Because of Lemma 5.8, $\forall i \in \mathbb{N} : (\Longrightarrow_{\mathrm{R},i} \cap\, (\mathcal{GT}(\mathrm{cons}) \times \mathcal{GT}(\mathrm{cons}))) \ \subseteq \sim$ is sufficient for this. But this is implied by Claim 1.

2. Let $\sim$ be as in Lemma 4.4. By the Homomorphism-Theorem we only have to show $(\overset{\circledast}{\Longleftrightarrow} \cap\, (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))) \ \subseteq \ \sim$. By $\mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig})) \neq \emptyset$ and Corollary 5.5 the following claim is sufficient for this:
   <u>Claim 2:</u> $\forall i \in \mathbb{N} : (\Longrightarrow_{\mathrm{R},\omega+i} \cap\, (\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))) \ \subseteq \sim$
   <u>Proof of Claim 2:</u> <u>$i = 0$:</u> By Claim 1.
   <u>$i \Rightarrow (i+1)$:</u> Let $s, t \in \mathcal{GT}(\mathrm{sig})$ with $s \Longrightarrow_{\mathrm{R},\omega+i+1} t$. If $s \Longrightarrow_{\mathrm{R},\omega} t$, then we have $s \sim t$ by the case '$i = 0$'. Otherwise there are $((l,r),C) \in \mathrm{R}$; $\sigma \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig}))$ (by Corollary 5.5 again); $p \in \mathcal{O}(s)$; with $s/p = l\sigma$; $t = s[\,p \leftarrow r\sigma\,]$; and $C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\mathrm{R},\omega+i}$. As $\mathcal{B}$ is a model of R, for $s \sim t$ (by the Substitution-Lemma(2.4)) we only have to show that the condition $C\sigma$ is true w. r. t. $\mathcal{B}$. Three cases for L in $C\sigma$: If $L = (\langle u,v\rangle, =)$, by $u \downarrow_{\mathrm{R},\omega+i} v$ the induction hypothesis implies $u \sim v$. If $L = (\langle u\rangle, \mathrm{Def})$, by $\exists \hat{u} \in \mathcal{GT}(\mathrm{cons}) : u \overset{\circledast}{\Longrightarrow}_{\mathrm{R},\omega+i} \hat{u}$ the induction hypothesis implies $\exists \hat{u} \in \mathcal{GT}(\mathrm{cons}) : u \sim \hat{u}$. If $L = (\langle u,v\rangle, \neq)$, by the existence of some $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$ with $u \overset{\circledast}{\Longrightarrow}_{\mathrm{R},\omega+i} \hat{u} \not\downarrow_{\mathrm{R},\omega+i} \hat{v} \overset{\circledast}{\Longleftarrow}_{\mathrm{R},\omega+i} v$ the induction hypothesis implies $u \sim \hat{u} \not\downarrow_{\mathrm{R},\omega+i} \hat{v} \sim v$. Thus, for $u \not\sim v$ it is sufficient to show $\hat{u} \not\sim \hat{v}$. By Lemma 5.9 we know that $\hat{u} \not\downarrow \hat{v}$ and therefore (by confluence of $\Longrightarrow$) $\hat{u} \overset{\circledast}{\not\Longleftrightarrow} \hat{v}$. Now we have to make a case distinction according to our two disjunctive premises in the theorem:

   (a) Because of (*1*), $\mathcal{B}$ must be not only a constructor-minimal model but also a constructor-minimum model of R. Hence $\hat{u} \not\sim \hat{v}$ simply because $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$; $\hat{u} \overset{\circledast}{\not\Longleftrightarrow} \hat{v}$; and $\mathcal{GT}(\mathrm{sig})/\overset{\circledast}{\Longleftrightarrow}$ is (by (*1*)) a model of R.

   (b) Simply because $\hat{u} \overset{\circledast}{\not\Longleftrightarrow} \hat{v}$.

   <u>Q. e. d. (Claim 2)</u>

<u>Q. e. d. (Theorem 5.14)</u>

Proof of Theorem 5.15

1. By Lemma 5.7 it suffices to show:
   Claim 1: $\forall i \in \mathbb{N} : \forall n \in \mathbb{N} : \forall s \in \mathcal{T}(\text{cons}) : \forall t : \left(s \overset{n}{\Longrightarrow}_{\text{R},i} t \quad\Leftrightarrow\quad s \overset{n}{\Longrightarrow}_{\text{R}',i} t \right)$
   Proof of Claim 1: $i = 0$: $\Longrightarrow_{\text{R},0} = \emptyset = \Longrightarrow_{\text{R}',0}$.
   $i \Rightarrow (i+1)$: $n = 0$: Trivial.
   $n \Rightarrow (n+1)$: "$\Leftarrow$": Suppose $s \overset{n}{\Longrightarrow}_{\text{R}',i+1} s' \Longrightarrow_{\text{R}',i+1} t$. By induction hypothesis in n we know $s \overset{n}{\Longrightarrow}_{\text{R},i+1} s'$ and hence by Lemma 5.7 $s' \in \mathcal{T}(\text{cons})$. By ($) there must be some $l \in \mathcal{T}(\text{cons})$; $((l,r),C) \in \text{R}$; $\sigma \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{cons}))$; $p \in \mathcal{O}(s')$ with $s'/p = l\sigma$; $t = s'[\,p \leftarrow r\sigma\,]$; $C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\text{R}',i}$. By the structure of constructor equations we have no literal in $C\sigma$ of the form $(e, \neq)$. For $(\langle u,v\rangle, =)$ in $C\sigma$ we have $u {\downarrow_{\text{R}',i}} v$ and $u, v \in \mathcal{T}(\text{cons})$ and therefore by induction hypothesis in i: $u {\downarrow_{\text{R},i}} v$.
   For $(\langle u\rangle, \text{Def})$ in $C\sigma$ we have $u \overset{\circledast}{\Longrightarrow}_{\text{R}',i} \hat{u}$ and $u \in \mathcal{T}(\text{cons})$ for some $\hat{u} \in \mathcal{GT}(\text{cons}')$ and hence by induction hypothesis in i: $u \overset{\circledast}{\Longrightarrow}_{\text{R},i} \hat{u}$, which also means $\hat{u} \in \mathcal{GT}(\text{cons})$.
   Thus, finally we get $C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\text{R}',i+1}$, i. e. $s' \Longrightarrow_{\text{R},i+1} t$.
   "$\Rightarrow$": Analogously.
   Q. e. d. (Claim 1)

2. It suffices to show $\forall i \in \mathbb{N} : (\Longrightarrow_{\text{R},\omega+i} \;\subseteq\; \Longrightarrow_{\text{R}'})$.
   $i = 0$: By (*1*).
   $i \Rightarrow (i+1)$: Suppose $s \Longrightarrow_{\text{R},\omega+i+1} t$. If $s \Longrightarrow_{\text{R},\omega} t$ we have $s \Longrightarrow_{\text{R}'} t$ by the case '$i = 0$'. Otherwise there must be some $((l,r),C) \in \text{R}$; $\sigma \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{sig}))$; $p \in \mathcal{O}(s)$ with $s/p = l\sigma$; $t = s[\,p \leftarrow r\sigma\,]$; $C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\text{R},\omega+i}$. The only thing to be shown is: $C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\text{R}'}$. For $(\langle u,v\rangle, =)$ in $C\sigma$ we have $u {\downarrow_{\text{R},\omega+i}} v$ and therefore by induction hypothesis $u {\downarrow_{\text{R}'}} v$. For $(\langle u\rangle, \text{Def})$ in $C\sigma$ we have $u \overset{\circledast}{\Longrightarrow}_{\text{R},i} \hat{u}$ for some $\hat{u} \in \mathcal{GT}(\text{cons})$ and therefore by induction hypothesis $u \overset{\circledast}{\Longrightarrow}_{\text{R}'} \hat{u}$. For $(\langle u,v\rangle, \neq)$ in $C\sigma$ we have $u \overset{\circledast}{\Longrightarrow}_{\text{R},\omega+i} \hat{u} \not\downarrow_{\text{R},\omega+i} \hat{v} \overset{\circledast}{\Longleftarrow}_{\text{R},\omega+i} v$ for some $\hat{u}, \hat{v} \in \mathcal{GT}(\text{cons})$. By Lemma 5.9 we get $\hat{u} \not\downarrow_{\text{R}} \hat{v}$ and then by (*1*): $\hat{u} \not\downarrow_{\text{R}'} \hat{v}$. Finally, by induction hypothesis we get $u \overset{\circledast}{\Longrightarrow}_{\text{R}'} \hat{u} \not\downarrow_{\text{R}'} \hat{v} \overset{\circledast}{\Longleftarrow}_{\text{R}'} v$.

Proof of Lemma 6.2

For $(((t_0,t_1),D),\hat{t},p) \in \text{CP(R)}$ there are $((l_k,r_k),C_k) \in \text{R}$; $k < 2$; $\xi, \sigma \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{sig}))$ with $l_0\xi\sigma = l_1\sigma/p$; $((t_0,t_1),D) = ((l_1[\,p \leftarrow r_0\xi\,], r_1), C_0\xi\, C_1)\sigma$. Let $\varphi \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{sig})[\cap\mathcal{GT}(\text{sig})])$ and assume $D\varphi$ to be fulfilled. Then $C_0\xi\sigma\varphi$ and $C_1\sigma\varphi$ are fulfilled, too. Therefore: $t_0\varphi = l_1\sigma\varphi[\,p \leftarrow r_0\xi\sigma\varphi\,] \Longleftarrow l_1\sigma\varphi \Longrightarrow r_1\sigma\varphi = t_1\varphi$ and by [ground] confluence we have $t_0\varphi \downarrow t_1\varphi$. Q. e. d. (Lemma 6.2)

Proof of Lemma 6.3

1. For $p = \emptyset$ the definition of 'quasi overlay [ground] joinable' reads
   $$\forall\varphi \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{sig})[\cap\mathcal{GT}(\text{sig})]) : (\; D\varphi \text{ fulfilled } \Rightarrow\; t_0\varphi \downarrow t_1\varphi \overset{\oplus}{\Longleftarrow} \hat{t}\varphi)$$
   As we assume $(((t_0,t_1),D),\hat{t},\emptyset)$ to be [ground] joinable, we now only have to show $t_1\varphi \overset{\oplus}{\Longleftarrow} \hat{t}\varphi$ under the assumption of $D\varphi$ being fulfilled. There are $((l_k,r_k),C_k) \in \text{R}$; $k < 2$; $\xi, \sigma \in \mathcal{SUB}(\text{V}, \mathcal{T}(\text{sig}))$ with $(((\ldots,t_1),D),\hat{t}) = (((\ldots,r_1),C_0\xi\, C_1),l_1)\sigma$. Since $C_1\sigma\varphi$ is fulfilled, we have $t_1\varphi = r_1\sigma\varphi \Longleftarrow l_1\sigma\varphi = \hat{t}\varphi$.

2. If $D\varphi$ is fulfilled, then we get by quasi overlay [ground] joinability and Corollary 5.4: $t_0\varphi = t_0\varphi[\,p \leftarrow t_0\varphi/p\,] \downarrow t_0\varphi[\,p \leftarrow t_1\varphi/p\,] = t_1\varphi$.

Q. e. d. (Lemma 6.3)

Proof of Theorem 6.4

We do the proof only for the case without the optional '[ground]', because the ground-case is shown perfectly analogously using Lemma 5.11.

For the proof of Subclaim 3 below, we extend the signatures by a new sort s_{new} and new constructor symbols eq_s for each old sort $s \in \mathrm{S}$ with arity $ss \longrightarrow s_{\mathrm{new}}$ and $\bot$ with arity s_{new}. We take (in addition to R) the following set of new rules:

$$\mathrm{R}' := \{\ \mathsf{eq}_s\, x\, x = \bot \mid s \in \mathrm{S} \ \wedge\ x = \min \mathrm{V}_s\ \}$$

Now by Theorem 5.15 we know that $\Longrightarrow$ is monotonic w. r. t. this extension. Therefore (as no new critical peaks occur) the critical peaks stay quasi overlay joinable. We are going to show the confluence of $\Longrightarrow_{\mathrm{R}\cup\mathrm{R}'}$, which implies the confluence of $\Longrightarrow_{\mathrm{R/sig/cons/V}}$ (defined over the non-enriched signatures) because the sort restrictions do not allow $\Longrightarrow_{\mathrm{R}\cup\mathrm{R}'}$ to make any use of terms of the sort s_{new} when rewriting terms of an 'old' sort. By the same argument we know that $\Longrightarrow_{\mathrm{R}\cup\mathrm{R}'}$ is noetherian on each 'old' sort; and by this we also know $\Longrightarrow_{\mathrm{R}\cup\mathrm{R}'}$ to be noetherian on s_{new} because $\bot$ is irreducible and terms of the form '$\mathsf{eq}_s\, u\, v$' allow at most one reduction step via $\Longrightarrow_{\mathrm{R}\cup\mathrm{R}'} \setminus \Longrightarrow_{\mathrm{R/sig'/cons'/V'}}$.

We define $\Longrightarrow_0 := \Longrightarrow_{\mathrm{R}'}$; $\Longrightarrow_\beta := \Longrightarrow_{\mathrm{R}\cup\mathrm{R}',\beta}$ for any ordinal β with $0 \prec \beta \prec \omega + \omega$ (where $\prec$ is the ordering of ordinal numbers); and $\Longrightarrow := \Longrightarrow_{\omega+\omega} := \Longrightarrow_{\mathrm{R}\cup\mathrm{R}'}$.

For $v, u, s, t \in \mathcal{T}(\mathrm{sig})$ with $v \overset{\circledast}{\Longleftarrow} u$ and $s \overset{\circledast}{\Longrightarrow} t$; $\Pi \subseteq \mathcal{O}(u)$ with $\forall p, q \in \Pi : (p \neq q \ \Rightarrow\ p|q)$ and $\forall o \in \Pi : u/o = s$; we say that $P(v, u, s, t, \Pi)$ holds <u>:iff</u> $v \downarrow u[\, o \leftarrow t \mid o \in \Pi]$. Now (by $\Pi := \{\emptyset\}$) it suffices to show that $P(v, u, s, t, \Pi)$ holds for all appropriate v, u, s, t, Π. We will show this by noetherian induction over the lexicographic combination of the following orderings:

1. $(\overset{\oplus}{\Longrightarrow}_{\mathrm{R}\cup\mathrm{R}'} \cup \rhd_{\mathrm{ST}})^{\oplus}$ (Cf. Lemma 7.5)
2. $\succ$
3. $\succ$

using the following measure on (v, u, s, t, Π):

1. s
2. the smallest ordinal $\beta \prec \omega + \omega + 1$ for which $v \overset{\circledast}{\Longleftarrow}_\beta u$
3. the smallest $n \in \mathbb{N}$ for which $v \overset{n}{\Longleftarrow}_\beta u$ for the β of (2)

For the limit ordinals $0, \omega, \omega + \omega$ in the second position of the measure, the induction step is trivial ($\overset{\circledast}{\Longleftarrow}_0 \subseteq \Longleftarrow_0 \cup \mathrm{id}$ and $\Longrightarrow_0$ is a unique relation; $\overset{\circledast}{\Longleftarrow}_\omega \subseteq \bigcup_{i \in \mathbb{N}_+} \overset{\circledast}{\Longleftarrow}_{\mathrm{R}\cup\mathrm{R}',i}$; $\overset{\circledast}{\Longleftarrow}_{\omega+\omega} \subseteq \bigcup_{i \in \mathbb{N}} \overset{\circledast}{\Longleftarrow}_{\mathrm{R}\cup\mathrm{R}',\omega+i}$). Thus, as we now suppose a smallest (v, u, s, t, Π) with $P(v, u, s, t, \Pi)$ not holding for, the second position of the measure must be one of the non-limit ordinals $i + 1$ or $\omega + i + 1$ for some $i \in \mathbb{N}$. Since the proofs for these two cases are very similar, we do them in parallel by writing $\beta + 1$ in both cases.

As $P(v, u, s, t, \Pi)$ holds trivially for $u = v$ or $s = t$ we have some u', s' with $v \overset{n}{\Longleftarrow}_{\beta+1} u' \Longleftarrow_{\beta+1} u$ and $s \Longrightarrow s' \overset{\circledast}{\Longrightarrow} t$. Now for a contradiction it is sufficient to show

<u>**Claim**</u>: There is some z with $v \overset{\circledast}{\Longrightarrow} z \overset{\circledast}{\Longleftarrow} u[\, o \leftarrow s' \mid o \in \Pi]$.

because then we have $z \downarrow u[\, o \leftarrow t \mid o \in \Pi]$ by $P(z, u[\, o \leftarrow s' \mid o \in \Pi], s', t, \Pi)$, which is smaller than (v, u, s, t, Π) in the first position of the measure by $s \overset{\oplus}{\Longrightarrow}_{\mathrm{R}\cup\mathrm{R}'} s'$.

If we had $s \Longrightarrow s'$ by some redex below the top of s, then **Claim** would hold by induction on the first position using $\rhd_{\mathrm{ST}}$. Thus, there are some $((l_0, r_0), C_0) \in \mathrm{R} \cup \mathrm{R}'$; $\mu_0 \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$; with $s = l_0\mu_0$; $s' = r_0\mu_0$; $C_0\mu_0$ fulfilled. Furthermore, we have some $q \in \mathcal{O}(u)$; $((l_1, r_1), C_1) \in \mathrm{R} \cup \mathrm{R}'$; $\mu_1 \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$; with $u/q = l_1\mu_1$; $u' = u[\, q \leftarrow r_1\mu_1\,]$; $C_1\mu_1$ fulfilled w. r. t. $\Longrightarrow_\beta$.

We now distinguish two cases by the relative position of q and Π:

“q not strictly below any $p \in \Pi$”: There is no p' with $pp' = q$, $p' \neq \emptyset$, and $p \in \Pi$

Define $\Pi' := \{\ p \mid qp \in \Pi\ \}$.

The critical peak case: There is some $p \in \Pi' \cap \mathcal{O}(l_1)$ with $l_1/p \notin \mathrm{V}$.

Let $\xi := \min \mathrm{Sep}(\mathcal{V}(((l_0, r_0), C_0)), \mathcal{V}(((l_1, r_1), C_1)))$. Let ϱ be given by $x\varrho = \left\{\begin{matrix} x\mu_1 & \text{if } x \in \mathcal{V}(((l_1, r_1), C_1)) \\ x\xi^{-1}\mu_0 & \text{otherwise} \end{matrix}\right\}$ $(x \in \mathrm{V})$. By $l_0\xi\varrho = l_0\xi\xi^{-1}\mu_0 = s = u/qp = l_1\mu_1/p = (l_1/p)\varrho$ let $\sigma := \min \mathrm{Mgu}(l_0\xi, l_1/p)$ and $\varphi \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$ with $\sigma\varphi = \varrho$. Let $(((t_0, t_1), D), \hat{t}) := (((l_1[\,p \leftarrow r_0\xi\,], r_1), C_0\xi\, C_1), l_1)\sigma$. Now we have $(((t_0, t_1), D), \hat{t})\varphi = (((l_1\mu_1[\,p \leftarrow r_0\mu_0\,], r_1\mu_1), C_0\mu_0\, C_1\mu_1), l_1\mu_1)$. Therefore $D\varphi$ is fulfilled.

Subclaim 1: There is some w with $s \overset{\oplus}{\Longrightarrow} r_1\mu_1/p \overset{\circledast}{\Longrightarrow} w \overset{\circledast}{\Longleftarrow} s'$ and $r_1\mu_1 = l_1\mu_1[\,p \leftarrow r_1\mu_1/p\,]$.

Proof of Subclaim 1: If $t_0 = t_1$, then $s \Longrightarrow s' = r_0\mu_0 = t_0\varphi/p = t_1\varphi/p = r_1\mu_1/p$; and $r_1\mu_1 = r_1\mu_1[\,p \leftarrow r_1\mu_1/p\,] = t_1\varphi[\,p \leftarrow r_1\mu_1/p\,] = t_0\varphi[\,p \leftarrow r_1\mu_1/p\,] = l_1\mu_1[\,p \leftarrow r_1\mu_1/p\,]$. Otherwise, if $t_0 \neq t_1$, $(((t_0, t_1), D), \hat{t}, p)$ is a critical peak and we get $s = u/qp = l_1\mu_1/p = \hat{t}\varphi/p \overset{\oplus}{\Longrightarrow} t_1\varphi/p \downarrow (t_0/p)\varphi = r_0\mu_0 = s'$ where $t_1\varphi/p = r_1\mu_1/p$; and $r_1\mu_1 = t_1\varphi = t_0\varphi[\,p \leftarrow t_1\varphi/p\,] = l_1\mu_1[\,p \leftarrow r_1\mu_1/p\,]$. Q. e. d. (Subclaim 1)

Subclaim 2: For $o \in \Pi \setminus \{qp\}$ we have $u'/o = s$.

Proof of Subclaim 2: If there is some p' with $o = qp'$, then by Subclaim 1: $u'/o = u'/qp' = r_1\mu_1/p' = l_1\mu_1[\,p \leftarrow \ldots\,]/p' = l_1\mu_1/p' = u/qp' = u/o = s$. Otherwise, by $qp \in \Pi$ we know $o|q$, and therefore $u'/o = u/o = s$. Q. e. d. (Subclaim 2)

By Subclaim 1 we get $u'[\,qp \leftarrow w\,] = u[\,q \leftarrow r_1\mu_1[\,p \leftarrow w\,]\,] = u[\,q \leftarrow l_1\mu_1[\,p \leftarrow w\,]\,] = u[\,qp \leftarrow w\,]$. Hence, for $\hat{u} := u'[\,qp \leftarrow w\,][\,o \leftarrow s' \mid o \in \Pi \setminus \{qp\}]$ we get $\hat{u} = u[\,qp \leftarrow w\,][\,o \leftarrow s' \mid o \in \Pi \setminus \{qp\}] \overset{\circledast}{\Longleftarrow} u[\,o \leftarrow s' \mid o \in \Pi]$ by $w \overset{\circledast}{\Longleftarrow} s'$ (by Subclaim 1). Thus, for **Claim** we only have to show $v \downarrow \hat{u}$. For $u'' := u'[\,o \leftarrow s' \mid o \in \Pi \setminus \{qp\}]$ (Cf. Subclaim 2) there is some w' with $v \overset{\circledast}{\Longrightarrow} w' \overset{\circledast}{\Longleftarrow} u''$ by Subclaim 2 and $P(v, u', s, s', \Pi \setminus \{qp\})$, which is smaller in the second or third position. Finally, we get $w' \downarrow \hat{u}$ by $u''/qp = u'/qp = r_1\mu_1/p$, Subclaim 1, and $P(w', u'', r_1\mu_1/p, w, \{qp\})$, which is smaller in the first position by Subclaim 1. Q. e. d. (The critical peak case)

The variable overlap (if any) case: $\forall p \in \Pi' \cap \mathcal{O}(l_1) : l_1/p \in \mathrm{V}$

Define $\Gamma : \mathrm{V} \to \mathcal{F}(\mathrm{O}[\mathcal{T}(\mathrm{sig})])$ by $(x \in \mathrm{V})$: $\Gamma(x) := \{\ p' \mid \exists p : (l_1/p = x\ \wedge\ pp' \in \Pi')\ \}$.

Define μ_1' by $(x \in \mathrm{V})$: $x\mu_1' := x\mu_1[\,p' \leftarrow s' \mid p' \in \Gamma(x)]$. Define $\hat{u} := u[\,q \leftarrow r_1\mu_1'\,][\,o \leftarrow s' \mid o \in \Pi \setminus (q\Pi')]$ and $\check{u} := u[\,q \leftarrow l_1\mu_1'\,][\,o \leftarrow s' \mid o \in \Pi \setminus (q\Pi')]$.

We are going to show $v \downarrow \hat{u} \Longleftarrow \check{u} \overset{\circledast}{\Longleftarrow} u[\,o \leftarrow s' \mid o \in \Pi]$ for **Claim**. Since for $p' \in \Gamma(x)$ we always have some p with $l_1/p = x$; $x\mu_1/p' = l_1\mu_1/pp' = u/qpp' = s \Longrightarrow s'$; we get $v \downarrow \hat{u}$ by $P(v, u', s, s', (\Pi \setminus (q\Pi')) \cup \{\ qpp' \mid \exists x : (r_1/p = x\ \wedge\ p' \in \Gamma(x))\ \})$, which is smaller in the second or third position. We get $\hat{u} \Longleftarrow \check{u}$ by

Subclaim 3: $C_1\mu_1'$ is fulfilled.

Finally, $\check{u} \overset{\circledast}{\Longleftarrow} u[\,q \leftarrow l_1\mu_1[\,o \leftarrow s' \mid o \in \Pi']\,][\,o \leftarrow s' \mid o \in \Pi \setminus (q\Pi')] = u[\,o \leftarrow s' \mid o \in \Pi]$.

Proof of Subclaim 3: For $(\langle \bar{u}, \bar{v}\rangle, =)$ in C_1 we have $\bar{u}\mu_1 {\downarrow_\beta} \bar{v}\mu_1$ and hence for the sort $\ddot{u}$ of $\bar{u}$: $\bot \overset{\oplus}{\Longleftarrow}_\beta (\mathsf{eq}_{\ddot{u}}\,\bar{u}\,\bar{v})\mu_1$. We get $\bot \downarrow (\mathsf{eq}_{\ddot{u}}\,\bar{u}\,\bar{v})\mu_1'$ by $P(\bot,\ (\mathsf{eq}\,\bar{u}\,\bar{v})\mu_1,\ s,\ s'\ , \{\ pp' \mid \exists x : ((\mathsf{eq}_{\ddot{u}}\,\bar{u}\,\bar{v})/p = x\ \wedge\ p' \in \Gamma(x))\ \})$, which is smaller in the second position. Since there are no rules for $\bot$ and only one for $\mathsf{eq}_{\ddot{u}}$, this means $\bar{u}\mu_1' \downarrow \bar{v}\mu_1'$. For $(\langle \bar{u}\rangle, \mathrm{Def})$ in C_1 we know the existence of some $\vec{u} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_1 \overset{\circledast}{\Longrightarrow}_\beta \vec{u}$. We get some $\hat{u}$ with $\vec{u} \overset{\circledast}{\Longrightarrow} \hat{u} \overset{\circledast}{\Longleftarrow} \bar{u}\mu_1'$ by $P(\vec{u}, \bar{u}\mu_1, s, s', \{\ pp' \mid \exists x : (\bar{u}/p = x\ \wedge\ p' \in \Gamma(x))\ \})$, which is smaller in the second position. By Lemma 5.8 we get $\hat{u} \in \mathcal{GT}(\mathrm{cons})$. Finally, for $(\langle \bar{u}, \bar{v}\rangle, \neq)$ in C_1 we have some $\vec{u}, \vec{v} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_1 \overset{\circledast}{\Longrightarrow}_\beta \vec{u} \not\downarrow \vec{v} \overset{\circledast}{\Longleftarrow}_\beta \bar{v}\mu_1$. By applying the same procedure as before twice we get $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_1' \overset{\circledast}{\Longrightarrow} \hat{u} \overset{\circledast}{\Longleftarrow} \vec{u} \not\downarrow \vec{v} \overset{\circledast}{\Longrightarrow} \hat{v} \overset{\circledast}{\Longleftarrow} \bar{v}\mu_1'$, i. e. $\bar{u}\mu_1' \overset{\circledast}{\Longrightarrow} \hat{u} \not\downarrow \hat{v} \overset{\circledast}{\Longleftarrow} \bar{v}\mu_1'$.

Q. e. d. (Subclaim 3; The variable overlap (if any) case; “q not strictly below any $p \in \Pi$”)

"q strictly below $p \in \Pi$": There is some p' with $pp' = q$, $p' \neq \emptyset$, and $p \in \Pi$.
Subclaim 4: For $o \in \Pi \setminus \{p\}$ we have $u'/o = s = l_0\mu_0$.
Proof of Subclaim 4: Since $o|p$, we have $u'/o = u[\,pp' \leftarrow \ldots]/o = u/o = s = l_0\mu_0$.
Q. e. d. (Subclaim 4)
The (second) critical peak case: $p' \in \mathcal{O}(l_0) \;\wedge\; l_0/p' \notin \mathrm{V}$
Let $\xi := \min \mathrm{Sep}(\mathcal{V}(((l_1, r_1), C_1)), \mathcal{V}(((l_0, r_0), C_0)))$. Let ϱ be given by $x\varrho = \left\{\begin{matrix} x\mu_0 & \text{if } x \in \mathcal{V}(((l_0, r_0), C_0)) \\ x\xi^{-1}\mu_1 & \text{otherwise} \end{matrix}\right\}$ $(x \in \mathrm{V})$. By $l_1\xi\varrho = l_1\xi\xi^{-1}\mu_1 = u/q = u/pp' = l_0\mu_0/p' = (l_0/p')\varrho$ let $\sigma := \min \mathrm{Mgu}(l_1\xi, l_0/p')$ and $\varphi \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig}))$ with $\sigma\varphi = \varrho$. Let $(((t_0, t_1), D), \hat{t}) := (((l_0[\,p' \leftarrow r_1\xi\,], r_0), C_1\xi\, C_0), l_0)\sigma$. Now we have $(((t_0, t_1), D), \hat{t})\varphi = (((l_0\mu_0[\,p' \leftarrow r_1\mu_1\,], r_0\mu_0), C_1\mu_1\, C_0\mu_0), l_0\mu_0)$. Therefore $D\varphi$ is fulfilled.
Subclaim 5: There is some w with $s \Longrightarrow s[\,p' \leftarrow r_1\mu_1\,] \stackrel{\circledast}{\Longrightarrow} w \stackrel{\circledast}{\Longleftarrow} s'$.
Proof of Subclaim 5: Since $s/p' = u/pp' = u/q = l_1\mu_1$ we have $s \Longrightarrow s[\,p' \leftarrow r_1\mu_1\,]$. Because $s[\,p' \leftarrow r_1\mu_1\,] = l_0\mu_0[\,p' \leftarrow r_1\mu_1\,] = t_0\varphi$ and $t_1\varphi = r_0\mu_0 = s'$ we now only have to show $t_0\varphi \downarrow t_1\varphi$. If $t_0 = t_1$, this is trivial. Otherwise, if $t_0 \neq t_1$, $(((t_0, t_1), D), \hat{t}, p')$ is a critical peak and we get $t_0\varphi \downarrow t_1\varphi$ by Lemma 6.3. Q. e. d. (Subclaim 5)

For $\hat{u} := u[\,p \leftarrow w\,][\,o \leftarrow s' \mid o \in \Pi \setminus \{p\}]$ we get $\hat{u} \stackrel{\circledast}{\Longleftarrow} u[\,o \leftarrow s' \mid o \in \Pi]$ by $w \stackrel{\circledast}{\Longleftarrow} s'$ (by Subclaim 5). Thus, for **Claim** we only have to show $v \downarrow \hat{u}$. For $u'' := u'[\,o \leftarrow s' \mid o \in \Pi \setminus \{p\}]$ (Cf. Subclaim 4) there is some w' with $v \stackrel{\circledast}{\Longrightarrow} w' \stackrel{\circledast}{\Longleftarrow} u''$ by Subclaim 4 and $P(v, u', s, s', \Pi \setminus \{p\})$, which is smaller in the second or third position. Finally, we get $w' \downarrow \hat{u}$ by $u''/p = u'/p = u[\,pp' \leftarrow r_1\mu_1\,]/p = u/p[\,p' \leftarrow r_1\mu_1\,] = s[\,p' \leftarrow r_1\mu_1\,]$, Subclaim 5, and $P(w', u'', s[\,p' \leftarrow r_1\mu_1\,], w, \{p\})$, which is smaller in the first position by Subclaim 5.
Q. e. d. (The (second) critical peak case)
The (second) variable overlap case: There are $\check{p}, \hat{p}, x$ with $p' = \check{p}\hat{p}$ and $l_0/\check{p} = x \in \mathrm{V}$
Define μ_0' by $(y \in \mathrm{V})$: $y\mu_0' := \left\{\begin{matrix} x\mu_0[\,\hat{p} \leftarrow r_1\mu_1\,] & \text{if } y = x \\ y\mu_0 & \text{otherwise} \end{matrix}\right\}$. Define $\hat{u} := u[\,o \leftarrow r_0\mu_0' \mid o \in \Pi]$ and $\check{u} := u[\,o \leftarrow l_0\mu_0' \mid o \in \Pi]$. Since $x\mu_0/\hat{p} = l_0\mu_0/\check{p}\hat{p} = s/p' = u/pp' = u/q = l_1\mu_1 \Longrightarrow r_1\mu_1$ and $u' = u[\,p\check{p}\hat{p} \leftarrow r_1\mu_1\,] = u[\,p \leftarrow l_0\mu_0[\,\check{p} \leftarrow x\mu_0'\,]\,][\,o \leftarrow l_0\mu_0 \mid o \in \Pi \setminus \{p\}]$ (by Subclaim 4), we get some w with $v \stackrel{\circledast}{\Longrightarrow} w \stackrel{\circledast}{\Longleftarrow} \check{u}$ by $P(v, u', l_1\mu_1, r_1\mu_1, \{\, o\check{o}\hat{p} \mid o \in \Pi \,\wedge\, l_0/\check{o} = x \,\wedge\, o\check{o} \neq p\check{p}\,\})$, which is smaller in the first position by $s \rhd_{\mathrm{ST}} s/p' = u/pp' = u/q = l_1\mu_1$. Now by $\hat{u} \stackrel{\circledast}{\Longleftarrow} u[\,o \leftarrow r_0\mu_0 \mid o \in \Pi] = u[\,o \leftarrow s' \mid o \in \Pi]$ for **Claim** we only have to show $w \downarrow \hat{u}$. But this is given by Subclaim 6 below and $P(w, \check{u}, l_0\mu_0', r_0\mu_0', \Pi)$, which is smaller in the first position of the measure by $s = l_0\mu_0 \stackrel{\oplus}{\Longrightarrow} l_0\mu_0'$.
Subclaim 6: $C_0\mu_0'$ is fulfilled.
Proof of Subclaim 6: For $(\langle\bar{u}, \bar{v}\rangle, =)$ in C_0 we have some w with $\bar{u}\mu_0 \stackrel{\circledast}{\Longrightarrow} w \stackrel{\circledast}{\Longleftarrow} \bar{v}\mu_0$. We get some w' with $w \stackrel{\circledast}{\Longrightarrow} w' \stackrel{\circledast}{\Longleftarrow} \bar{u}\mu_0'$ by $P(w, \bar{u}\mu_0, l_1\mu_1, r_1\mu_1, \{\, o\hat{p} \mid \bar{u}/o = x\,\})$, and then $w' \downarrow \bar{v}\mu_0'$ by $P(w', \bar{v}\mu_0, l_1\mu_1, r_1\mu_1, \{\, o\hat{p} \mid \bar{v}/o = x\,\})$, which are smaller in the first position of the measure by $s \rhd_{\mathrm{ST}} l_1\mu_1$ as before. For $(\langle\bar{u}\rangle, \mathrm{Def})$ in C_0 we know the existence of some $\vec{u} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_0 \stackrel{\circledast}{\Longrightarrow} \vec{u}$. We get some $\hat{u}$ with $\vec{u} \stackrel{\circledast}{\Longrightarrow} \hat{u} \stackrel{\circledast}{\Longleftarrow} \bar{u}\mu_0'$ by $P(\vec{u}, \bar{u}\mu_0, l_1\mu_1, r_1\mu_1, \{\, o\hat{p} \mid \bar{u}/o = x\,\})$, which is smaller in the first position (as before). By Lemma 5.8 we get $\hat{u} \in \mathcal{GT}(\mathrm{cons})$. Finally, for $(\langle\bar{u}, \bar{v}\rangle, \neq)$ in C_0 we have some $\vec{u}, \vec{v} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_0 \stackrel{\circledast}{\Longrightarrow} \vec{u} \not\downarrow \vec{v} \stackrel{\circledast}{\Longleftarrow} \bar{v}\mu_0$. By applying the same procedure as before twice we get $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$ with $\bar{u}\mu_0' \stackrel{\circledast}{\Longrightarrow} \hat{u} \stackrel{\circledast}{\Longleftarrow} \vec{u} \not\downarrow \vec{v} \stackrel{\circledast}{\Longrightarrow} \hat{v} \stackrel{\circledast}{\Longleftarrow} \bar{v}\mu_0'$, i. e. $\bar{u}\mu_0' \stackrel{\circledast}{\Longrightarrow} \hat{u} \not\downarrow \hat{v} \stackrel{\circledast}{\Longleftarrow} \bar{v}\mu_0'$.
Q. e. d. (Subclaim 6) Q. e. d. (The (second) variable overlap case)
Q. e. d. ("q strictly below $p \in \Pi$")
Q. e. d. (**Proof of Theorem 6.4**)

Proof of Lemma 7.1 and Lemma 7.2

It is standard to encode a universal deterministic Turing machine by a finite set of left-linear, non-overlapping rules. This can be done in the following way, where stop, left, right, nil, $\mathsf{0}$, $\perp$ are constant symbols, s is a unary function symbol, c and nth are binary, cmd, state are ternary, and T is sixary; and '$\mathsf{T}\ l\ a\ r\ c\ s\ p$' encodes the Turing machine with

$$\begin{array}{ll}
l & \text{being the tape to the left of the head} \\
a & \text{being the symbol under the head} \\
r & \text{being the tape to the right of the head} \\
c & \text{being the next command to be executed} \\
s & \text{being the next state to be in} \\
p & \text{being the program.}
\end{array}$$

The rules are:

$$\begin{array}{llllllllllllll}
\mathsf{T} & l & a & r & \mathsf{stop} & s & p = & \perp & & & & & & \\
\mathsf{T} & \mathsf{nil} & a & r & \mathsf{left} & s & p = & \mathsf{T} & \mathsf{nil} & \mathsf{0} & \mathsf{c}\,a\,r & \mathsf{cmd}\,\mathsf{0}\,s\,p & \mathsf{state}\,\mathsf{0}\,s\,p & p \\
\mathsf{T} & \mathsf{c}\,b\,l & a & r & \mathsf{left} & s & p = & \mathsf{T} & l & b & \mathsf{c}\,a\,r & \mathsf{cmd}\,b\,s\,p & \mathsf{state}\,b\,s\,p & p \\
\mathsf{T} & l & a & \mathsf{nil} & \mathsf{right} & s & p = & \mathsf{T} & \mathsf{c}\,a\,l & \mathsf{0} & \mathsf{nil} & \mathsf{cmd}\,\mathsf{0}\,s\,p & \mathsf{state}\,\mathsf{0}\,s\,p & p \\
\mathsf{T} & l & a & \mathsf{c}\,b\,r & \mathsf{right} & s & p = & \mathsf{T} & \mathsf{c}\,a\,l & b & r & \mathsf{cmd}\,b\,s\,p & \mathsf{state}\,b\,s\,p & p \\
\mathsf{T} & l & a & r & \mathsf{0} & s & p = & \mathsf{T} & l & \mathsf{0} & r & \mathsf{cmd}\,\mathsf{0}\,s\,p & \mathsf{state}\,\mathsf{0}\,s\,p & p \\
\mathsf{T} & l & a & r & \mathsf{s}\,x & s & p = & \mathsf{T} & l & \mathsf{s}\,x & r & \mathsf{cmd}\,\mathsf{s}\,x\,s\,p & \mathsf{state}\,\mathsf{s}\,x\,s\,p & p
\end{array}$$

with the following auxiliary functions:

$$\begin{array}{llllllll}
\mathsf{nth} & \mathsf{0} & \mathsf{c}\,a\,l & & = a & & & \\
\mathsf{nth} & \mathsf{s}\,x & \mathsf{c}\,a\,l & & = \mathsf{nth} & x & l & \\
\mathsf{cmd} & a & s & \mathsf{c}\,c\,d & = \mathsf{nth} & a & \mathsf{nth} & s\ \ c \\
\mathsf{state} & a & s & \mathsf{c}\,c\,d & = \mathsf{nth} & a & \mathsf{nth} & s\ \ d
\end{array}$$

We use $\perp$ instead of a reasonable output because we are interested in the halting problem only. We assume all function symbols to be constructor symbols.

From this system we are now going to construct our positive-conditional rule system by exchanging the recursive T-rules of the form '$\mathsf{T}\ l\ a\ r\ c\ s\ p = \mathsf{T}\ l'\ a'\ r'\ c'\ s'\ p'$' for rules of the form '$\mathsf{T}\ l\ a\ r\ c\ s\ p\ =\ \perp\ \longleftarrow\ \ \mathsf{T}\ l'\ a'\ r'\ c'\ s'\ p'\ =\ \perp$'.

Now the above Turing machine halts iff the ground term '$\mathsf{T}\ l\ a\ r\ c\ s\ p$' is reducible. Therefore (the halting problem being not co-semi-decidable) the reducibility of ground terms cannot be co-semi-decidable.

For Lemma 7.2 we now add the following new rule:

$$\mathsf{foreverp}\ l\ a\ r\ c\ s\ p = \mathsf{true}\ \longleftarrow\ \ \mathsf{T}\ l\ a\ r\ c\ s\ p \neq \perp,\ \mathrm{Def}\,\mathsf{T}\ l\ a\ r\ c\ s\ p,\ \mathrm{Def}\perp$$

Now '$\mathsf{foreverp}\ l\ a\ r\ c\ s\ p$' is reducible iff the above Turing machine does not halt. Therefore the reducibility of ground terms cannot be semi-decidable.

Proof of Lemma 7.3

1. The function g that firstly tests whether its single argument s is in the enumerable set of [ground] terms, secondly tries to compute $f(s)$, and thirdly (if s is a [ground] term and $f(s)$ is defined) is defined iff the test $f(s) = s$ succeeds, is defined exactly on the irreducible [ground] terms.

2. Claim 1: $Y_\beta(s) := \{\ t \mid s \Longrightarrow_{\mathrm{R},\beta} t\ \wedge\ s,t \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]\ \}$ and
$Z_\beta(s) := \{\ t \mid s \overset{\circledast}{\Longrightarrow}_{\mathrm{R},\beta} t\ \wedge\ s,t \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]\ \}$
are enumerable for all $\beta \prec \omega + \omega + 1$.
Now if Claim 1 holds, we can enumerate $Z_{\omega+\omega}(s)$ and simultaneously the irreducible [ground] terms until one term t occurs in both enumerations and we can return $f(s) := t$.
Proof of Claim 1: $s \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]$ is semi-decidable. Hence it suffices to show that $Y_\beta(s)$ is enumerable for all $s \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]$ [by Lemma 5.11]. The induction step for the limit ordinals $0, \omega, \omega+\omega$ is trivial ($Y_0(s) = \emptyset$; $Y_\omega(s) = \bigcup_{i\prec\omega} Y_i(s)$; $Y_{\omega+\omega}(s) = \bigcup_{i\prec\omega} Y_{\omega+i}(s)$). The induction step for the non-limit ordinals $\beta+1$ can be done the following way: For all rules $((l,r),C) \in \mathrm{R}$ and all $p \in \mathcal{O}(s)$ and all $\sigma \in \mathcal{SUB}(\mathcal{V}(((l,r),C)), \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})])$ with $s/p = l\sigma$ we test in an enumerative fashion whether $C\sigma$ is fulfilled w. r. t. $\Longrightarrow_{\mathrm{R},\beta}[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]$ and enumerate $s[\,p \leftarrow r\sigma\,]$ if the test succeeds. For $\omega \prec \beta+1$ we merge this enumeration with that of $Y_\omega(s)$. The test of "$C\sigma$ fulfilled w. r. t. $\Longrightarrow_{\mathrm{R},\beta}[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]$" can be semi-decided the following way: For $(\langle u,v\rangle, =)$ in $C\sigma$ we test the enumerable set $Z_\beta(u) \cap Z_\beta(v)$ for non-emptiness. For $(\langle u\rangle, \mathrm{Def})$ in $C\sigma$ we test the enumerable set $Z_\beta(u) \cap \mathcal{GT}(\mathrm{cons})$ for non-emptiness. For $(\langle u,v\rangle, \neq)$ in $C\sigma$ we test for the existence of $\hat{u} \in A(u)$ and $\hat{v} \in A(v)$ with $\hat{u} \neq \hat{v}$ (syntactically) for the enumerable sets $A(w) := Z_\beta(w) \cap (\mathcal{GT}(\mathrm{cons}) \setminus \mathrm{dom}(\Longrightarrow_{\mathrm{R}}))$. This last test succeeds only if $\exists \hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons}) : u \overset{\circledast}{\Longrightarrow}_{\mathrm{R},\beta} \hat{u} \not\downarrow_{\mathrm{R},\beta} \hat{v} \overset{\circledast}{\Longleftarrow}_{\mathrm{R},\beta} v$ holds. It also succeeds if this property holds because of $\forall s \in \mathcal{GT}(\mathrm{cons}) : \exists t : s \overset{\circledast}{\Longrightarrow}_{\mathrm{R}} t \notin \mathrm{dom}(\Longrightarrow_{\mathrm{R}})$, Lemma 5.8, and $\omega \prec \beta+1$.

Proof of Lemma 7.5

1. For $t \rhd_{\mathrm{ST}} t' > t''$ there is a $p \in \mathcal{O}(t);\ p \neq \emptyset$ with $t' = t/p$. By monotony we get $t = t[\,p \leftarrow t'\,] > t[\,p \leftarrow t''\,] \rhd_{\mathrm{ST}} t''$.

2. $$\begin{array}{llll} \rhd_{\mathrm{ST}} \circ \rhd_{\mathrm{ST}} & \subseteq \rhd_{\mathrm{ST}} & & \subseteq \succ \\ \rhd_{\mathrm{ST}} \circ (> \circ \trianglerighteq_{\mathrm{ST}}) & \subseteq\ > \circ \rhd_{\mathrm{ST}} \circ \trianglerighteq_{\mathrm{ST}} & \subseteq\ > \circ \trianglerighteq_{\mathrm{ST}} & \subseteq \succ \\ (> \circ \trianglerighteq_{\mathrm{ST}}) \circ \rhd_{\mathrm{ST}} & \subseteq\ > \circ \trianglerighteq_{\mathrm{ST}} & & \subseteq \succ \\ (> \circ \trianglerighteq_{\mathrm{ST}}) \circ (> \circ \trianglerighteq_{\mathrm{ST}}) & \subseteq\ > \circ > \circ \trianglerighteq_{\mathrm{ST}} \circ \trianglerighteq_{\mathrm{ST}} & \subseteq\ > \circ \trianglerighteq_{\mathrm{ST}} & \subseteq \succ \end{array}$$
Hence $\succ$ is transitive. By $> \subseteq \succ$ and $\rhd_{\mathrm{ST}} \subseteq \succ$ we get $(> \cup \rhd_{\mathrm{ST}})^{\oplus} \subseteq \succ$, and then by $(> \circ \trianglerighteq_{\mathrm{ST}}),\ \rhd_{\mathrm{ST}} \subseteq (> \cup \rhd_{\mathrm{ST}})^{\oplus}$ we get $\succ\ =\ (> \cup \rhd_{\mathrm{ST}})^{\oplus}$.

3. Suppose that $\succ$ is not noetherian. As $\rhd_{\mathrm{ST}}$ and $>$ are noetherian, there must be $r, s : \mathbb{N} \to \mathcal{T}(\mathrm{sig})$ with $\forall i \in \mathbb{N} : r_i \rhd_{\mathrm{ST}} s_i > r_{i+1}$. Then there is a $p : \mathbb{N} \to \mathbb{N}_+^+$ with $\forall i \in \mathbb{N} : r_i/p_i = s_i$. Define $t_n := r_0\,[\,p_0 \leftarrow r_1\,[\,p_1 \leftarrow r_2 \ldots [\,p_{n-1} \leftarrow r_n\,] \ldots]]$. Because of $r_i/p_i = s_i > r_{i+1}$ we get $t_n \in \mathcal{T}(\mathrm{sig})$ (as $>$ sort-invariant) and $t_n > t_{n+1}$ (as $>$ monotonic). This contradicts the well-foundedness of $>$.

   If $>$ is stable, additionally, then $\succ$ is stable too, because $\rhd_{\mathrm{ST}}$ is.

Now here is an example for $\succ$ not sort-invariant and not monotonic: Let A, B be two different sorts. Let $\alpha(\mathrm{a}) = A$, $\alpha(\mathrm{f}) = A \longrightarrow B$, $\alpha(\mathrm{g}) = A \longrightarrow A$. Define $> := \emptyset$. Then we have $\succ = \ \rhd_{\mathrm{ST}}$ and therefrom: fa $\rhd_{\mathrm{ST}}$ a (hence not sort-invariant); and ga $\rhd_{\mathrm{ST}}$ a but fga $\not\rhd_{\mathrm{ST}}$ fa (hence not monotonic).

4. Take the signature from the example in (3). $> := \{(\mathrm{a}, \mathrm{fa})\}$ is a monotonic (indeed!), well-founded ordering on $\mathcal{T}(\mathrm{sig})$ that is not sort-invariant. Now $\succ$ is not irreflexive: a $>$ fa $\rhd_{\mathrm{ST}}$ a .

   If one changes $\alpha(\mathrm{f})$ to be $\alpha(\mathrm{f}) = A \longrightarrow A$, then $>$ is a sort-invariant, well-founded ordering on $\mathcal{T}(\mathrm{sig})$ that is not monotonic.

Proof of Lemma 7.6

Suppose $\rhd$ is not noetherian. Then there is a $t : \mathbb{N} \to \mathcal{T}(\mathrm{sig})$ with

$$\forall i \in \mathbb{N} : \ t_i (> \cup \rhd_{\mathrm{ST}} \cup \hookrightarrow) \, t_{i+1} \ .$$

Define $k : \mathbb{N} \to \mathbb{N}$ by:

$$\begin{aligned} k_0 &= 0 \\ k_{i+1} &= 1 + \min\{ \ j \mid j \geq k_i \ \wedge \ t_j \hookrightarrow t_{j+1} \} \end{aligned}$$

The above minimum always exists because $(> \cup \rhd_{\mathrm{ST}})^{\oplus}$ is noetherian. We have $t_{k_i}((> \cup \rhd_{\mathrm{ST}})^{\circledast} \circ \hookrightarrow) \ t_{k_{i+1}}$. By $\rhd_{\mathrm{ST}} \circ > \ \subseteq \ > \circ \rhd_{\mathrm{ST}}$ we get $t_{k_i}(>^{\circledast} \circ \rhd_{\mathrm{ST}}{}^{\circledast} \circ \hookrightarrow) \, t_{k_{i+1}}$, and by $\rhd_{\mathrm{ST}} \circ \hookrightarrow \ \subseteq \ \hookrightarrow$ finally $t_{k_i}(>^{\circledast} \circ \hookrightarrow) \, t_{k_{i+1}}$.

Hence we have $t_{k_i}(> \cup \hookrightarrow)^{\oplus} \ t_{k_{i+1}}$, which contradicts the assertion that $(> \cup \hookrightarrow)^{\oplus}$ is noetherian.

An alternative proof (using the Axiom of Choice) can be done the following way: $(> \cup \hookrightarrow)$ *quasi-commutes* over $(> \cup \rhd_{\mathrm{ST}})$ in the sense of Bachmair&Dershowitz[2]. Now $(> \cup \rhd_{\mathrm{ST}} \cup \hookrightarrow)$ must be noetherian by Theorem 1 of Bachmair&Dershowitz[2].

Proof of Lemma 7.13

$(\Longrightarrow_{\mathrm{R}}[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]) \ \subseteq \ >$ is trivial by double induction on the construction of $\Longrightarrow_{\mathrm{R}}$ [using Corollary 5.5 to restrict the variables occurring only in the condition terms to ground term substitution]. Therefore $\Longrightarrow_{\mathrm{R}}[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]$ is noetherian. By Corollary 5.5 $\Longrightarrow_{\mathrm{R}}$ is noetherian, too.

Under the additional assumptions it is sufficient to show:
Claim 1: For all $s \in \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})]$: $\{ \ t \mid s \overset{\oplus}{\Longrightarrow}_{\mathrm{R}} t \ \}$ is a finite computable set.
Proof of Claim 1: We show this by noetherian induction on $\lhd$: Assume s to be $\lhd$-smallest with Claim 1 not holding for. As R and $\mathcal{O}(s)$ are finite, R is extra-variable free, and R is compatible with $(>, \ \rhd)$, there are only finitely many t with $s \Longrightarrow_{\mathrm{R}} t$ and these t satisfy $s > t$ and therefore $s \ \rhd t$. By induction hypothesis this means that $\{ \ t \mid s \overset{\oplus}{\Longrightarrow}_{\mathrm{R}} t \ \}$ must be finite. Furthermore, we can compute this set by computing $\{ \ t \mid s \Longrightarrow_{\mathrm{R}} t \ \}$ by checking the instantiated condition literals ...

1. ... (firstly) for being $\lhd$-smaller than s and (secondly) for being fulfilled, which is possible because the induction hypothesis is applicable to the instantiated condition terms and $\mathcal{GT}(\mathrm{cons})$ is a decidable subset of $\mathcal{T}(\mathrm{sig})$.

2. ... from left to right until one of them fails because then their instantiated terms are $\lhd$-smaller than s and the induction hypothesis is applicable.

Proof of Theorem 7.14

(1) $\Rightarrow$ *(2)*: By Lemma 6.2. Q. e. d. ((*1*) $\Rightarrow$ (*2*))

(2) $\Rightarrow$ *(1)*: First notice that the usual modularisation of the proof for the unconditional analogue of the theorem (by showing first that local confluence is guaranteed except for the cases that are matched by critical peaks (the so-called "critical pair lemma")) is not possible here because we need the confluence-property to hold for the condition terms even for the cases that are not matched by critical peaks. Now to the proof: Let s be minimal in $\rhd$ with the [ground] confluence-property (with s as a peak) not holding for. Because of $(\Longrightarrow[\cap(\mathcal{GT}(\mathrm{sig}) \times \mathcal{GT}(\mathrm{sig}))]) \subseteq \rhd$ (by Lemma 7.13), even the local [ground] confluence-property doesn't hold for s. Let $p, q \in \mathcal{O}(s)$; $t_0 \Longleftarrow_p s \Longrightarrow_q t_1$; $t_0 \not\downarrow t_1$. Now as one of p, q must be a prefix of the other, w. l. o. g. say that q is a prefix of p. As $s \trianglerighteq s/q$, by the minimality of s we have $q = \emptyset$. Now for $k < 2$ [using Corollary 5.5 to restrict the variables occurring only in the condition terms to ground term substitution] there must be $((l_k, r_k), C_k) \in \mathrm{R}$; $\mu_k \in \mathcal{SUB}(\mathrm{V}, \mathcal{GT}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})])$; with $C_k\mu_k$ fulfilled; $((l_k, r_k), C_k)\mu_k$ aligned with T; $s = l_1\mu_1$; $s/p = l_0\mu_0$; $t_0 = l_1\mu_1[\,p \leftarrow r_0\mu_0\,]$; $t_1 = r_1\mu_1$.

The inductive case: $p = q_0q_1$; $l_1/q_0 = x \in \mathrm{V}$:

Let ν be the substitution given by $y\nu = \left\{ \begin{array}{ll} x\mu_1[\,q_1 \leftarrow r_0\mu_0\,] & \text{if } y = x \\ y\mu_1 & \text{otherwise} \end{array} \right\}$ $(y \in \mathrm{V})$.

As $x\mu_1/q_1 = l_1\mu_1/q_0q_1 = s/p = l_0\mu_0$ we have $y\mu_1 \stackrel{\leq 1}{\Longrightarrow} y\nu$ for $y \in \mathrm{V}$, and by monotony:
$t_0 = l_1\mu_1[\,p \leftarrow r_0\mu_0\,] = l_1[\,q_0 \leftarrow x\nu\,][\,q' \leftarrow y\mu_1 \mid l_1/q' = y \in \mathrm{V} \ \wedge\ q' \neq q_0\,]$
$\stackrel{\circledast}{\Longrightarrow} l_1[\,q' \leftarrow y\nu \mid l_1/q' = y \in \mathrm{V}\,] = l_1\nu =: \hat{t}_0$;
$t_1 = r_1\mu_1 \stackrel{\circledast}{\Longrightarrow} r_1\nu =: \hat{t}_1$. It suffices to show $\hat{t}_0 \Longrightarrow \hat{t}_1$, which follows from the following
Claim: $C_1\nu$ is fulfilled.
Proof of Claim: For each L in C_1 we have to show that $L\nu$ is fulfilled.
$L = (\langle u, v\rangle, =)$: We know $u\nu \stackrel{\circledast}{\Longleftarrow} u\mu_1 \downarrow v\mu_1 \stackrel{\circledast}{\Longrightarrow} v\nu$. As $s = l_1\mu_1 \rhd u\mu_1$ we have $u\nu \downarrow v\mu_1 \stackrel{\circledast}{\Longrightarrow} v\nu$. As $s = l_1\mu_1 \rhd v\mu_1$ we have $u\nu \downarrow v\nu$.
$L = (\langle u\rangle, \mathrm{Def})$: We know the existence of $\hat{u} \in \mathcal{GT}(\mathrm{cons})$ with $u\nu \stackrel{\circledast}{\Longleftarrow} u\mu_1 \stackrel{\circledast}{\Longrightarrow} \hat{u}$. By Lemma 5.8 and Lemma 7.13 we can additionally assume $\hat{u}$ to be irreducible. By $s = l_1\mu_1 \rhd u\mu_1$ we get $u\nu \stackrel{\circledast}{\Longrightarrow} \hat{u}$.
$L = (\langle u, v\rangle, \neq)$: We know the existence of $\hat{u}, \hat{v} \in \mathcal{GT}(\mathrm{cons})$ with

$$u\nu \stackrel{\circledast}{\Longleftarrow} u\mu_1 \stackrel{\circledast}{\Longrightarrow} \hat{u} \not\downarrow \hat{v} \stackrel{\circledast}{\Longleftarrow} v\mu_1 \stackrel{\circledast}{\Longrightarrow} v\nu \ .$$

Just like above we can additionally assume $\hat{u}, \hat{v}$ to be irreducible. By $s = l_1\mu_1 \rhd u\mu_1, v\mu_1$ we get $u\nu \downarrow \hat{u}$; $\hat{v} \downarrow v\nu$; and hence $u\nu \stackrel{\circledast}{\Longrightarrow} \hat{u} \not\downarrow \hat{v} \stackrel{\circledast}{\Longleftarrow} v\nu$.
Q. e. d. (Claim; The inductive case)

The critical peak case: $p \in \mathcal{O}(l_1)$; $l_1/p \notin \mathrm{V}$:

Let $\xi := \min \mathrm{Sep}(\mathcal{V}(l_0), \mathcal{V}(l_1))$. Let ϱ be given by $x\varrho = \left\{ \begin{array}{ll} x\mu_1 & \text{if } x \in \mathcal{V}(l_1) \\ x\xi^{-1}\mu_0 & \text{otherwise} \end{array} \right\}$ $(x \in \mathrm{V})$.

By $l_0\xi\varrho = l_0\xi\xi^{-1}\mu_0 = s/p = l_1\mu_1/p = l_1\varrho/p = (l_1/p)\varrho$ let $\sigma := \min \mathrm{Mgu}(l_0\xi, l_1/p)$ and $\tau \in \mathcal{SUB}(\mathrm{V}, \mathcal{T}(\mathrm{sig})[\cap\mathcal{GT}(\mathrm{sig})])$ with $\sigma\tau = \varrho$.
Let $((l', r'), D) := ((l_1[\,p \leftarrow r_0\xi\,], r_1), C_0\xi\, C_1)\sigma$. Now we have: $((l', r'), D)\tau = ((l_1[\,p \leftarrow r_0\xi\,], r_1), C_0\xi\, C_1)\varrho = ((l_1\mu_1[\,p \leftarrow r_0\mu_0\,], r_1\mu_1), C_0\mu_0\, C_1\mu_1) = ((t_0, t_1), C_0\mu_0\, C_1\mu_1)$.
Hence $D\tau$ is fulfilled. Furthermore, for $u \in \mathcal{TERMS}(D\tau)$ we get $u \in \mathcal{TERMS}(C_0\mu_0\, C_1\mu_1)$, and then $u \lhd l_0\mu_0 \trianglelefteq_{\mathrm{ST}} s$ or $u \lhd l_1\mu_1 = s$, and therefore by induction hypothesis we know that $\Longrightarrow$ is [ground] confluent below u. If $t_0 = t_1$, then $t_0 \downarrow t_1$ trivially. Otherwise $(((l', r'), D), l_1\sigma, p) \in \mathrm{CP}(\mathrm{R})$ and then by (*2*) we get $t_0 \downarrow t_1$, too.
Q. e. d. (The critical peak case; (*2*) $\Rightarrow$ (*1*))

References

[1] Leo Bachmair. *Proof by Consistency in Equational Theories.* 3^{rd} LICS 1988, pp. 228-233.

[2] Leo Bachmair, Nachum Dershowitz. *Commutation, Transformation, and Termination.* Proc. of 8^{th} CADE 1986, LNCS 230. Springer-Verlag, Berlin 1986.

[3] Leo Bachmair, Harald Ganzinger. *On Restrictions of Ordered Paramodulation with Simplification.* Proc. of 10^{th} CADE 1990, LNAI 449. Springer-Verlag, Berlin 1990.

[4] Leo Bachmair, Harald Ganzinger. *Perfect Model Semantics for Logic Programs with Equality.* Proc. of 8^{th} Int. Conf. on Logic Programming, pp. 645-659, MIT Press, 1991.

[5] Klaus Becker. *Inductive Proofs in Specifications Parameterized by a Built-In Theory.* SEKI-Report SR-92-02, 1992; Fachbereich Informatik, Universität Kaiserslautern, Postfach 3049, W-6750 Kaiserslautern.

[6] Eddy Bevers, Johan Lewi. *Proof by Consistency in Conditional Equational Theories.* Report CW 102, Revised July 1990; Departement of Computer Science, K. U. Leuven.

[7] Susanne Biundo, Birgit Hummel, Dieter Hutter, Christoph Walter. *The Karlsruhe Induction Theorem Proving System.* Proc. of 8^{th} CADE 1986, LNCS 230. Springer-Verlag, Berlin 1986.

[8] Robert S. Boyer, J Strother Moore. *A Computational Logic.* Academic Press, 1979.

[9] Robert S. Boyer, J Strother Moore. *A Computational Logic Handbook.* Academic Press, 1988.

[10] François Bronsard, Uday S. Reddy. *Conditional Rewriting in Focus.* Departement of Computer Science, The University of Illinois at Urbana-Champaign, Urbana, IL 61801.

[11] Nachum Dershowitz. *Termination of Rewriting.* J. Symbolic Computation (1987) **3**, pp. 69-116.

[12] Nachum Dershowitz, Mitsuhiro Okada, G. Sivakumar. *Confluence of Conditional Rewrite Systems.* 9^{th} International Conference on Automated Deduction 1988. LNAI 310. Springer-Verlag, Berlin 1988.

[13] Nachum Dershowitz, Mitsuhiro Okada, G. Sivakumar. *Canonical Conditional Rewrite Systems.* Proc. of 9^{th} CADE 1989, LNAI 449. Springer-Verlag, Berlin 1990.

[14] Herbert B. Enderton. *A mathematical introduction to logic.* Academic Press, 2^{nd} printing 1973.

[15] Harald Ganzinger. *A Completion Procedure for Conditional Equations.* First international workshop on Conditional Term Rewriting Systems. LNCS 308. Springer-Verlag, Berlin 1988.

[16] Harald Ganzinger, Jürgen Stuber. *Inductive Theorem Proving by Consistency for First-Order Clauses.* In: Informatik-Festschrift zum 60. Geburtstag von Günter Hotz. Teubner Verlag. To appear, 1992.

[17] Richard Göbel. *A Completion Procedure for Generating Ground Confluent Term Rewriting Systems.* Dissertation, 1988, Universität Kaiserslautern, Fachbereich Informatik, Postfach 3049, W-6750 Kaiserslautern.

[18] Bernhard Gramlich. *Inductive Theorem Proving Using Refined Unfailing Completion Techniques.* SEKI-Report SR-89-14, 1989; Fachbereich Informatik, Universität Kaiserslautern, Postfach 3049, W-6750 Kaiserslautern.

[19] Gérard Huet, Jean-Marie Hullot. *Proofs by Induction in Equational Theories with Constructors.* 21^{th} FOCS 1980, pp. 96-107; also in: JCSS 25, 1982, pp. 293-266.

[20] Jean Pierre Jouannaud, Bernard Waldmann. *Reductive Conditional Term Rewriting Systems.* Conference on Formal Description of Programming Concepts III, 1986, pp. 223-244.

[21] Sam Kamin, Jean-Jacques Levy. *Attempts for Generalizing the Recursive Path Ordering.* Unpublished Manuscript, Departement of Computer Science, University of Illinois, Urbana, Illinois, 1980.

[22] Stéphane Kaplan. *Conditional Rewrite Rules.* Theoretical Computer Science 33 (1984), pp. 175-193. North-Holland.

[23] Stéphane Kaplan. *Positive/Negative Conditional Rewriting.* First international workshop on Conditional Term Rewriting Systems. LNCS 308. Springer-Verlag, Berlin 1988.

[24] Stéphane Kaplan, Marianne Choquer. *On the Decidability of Quasi-Reducibility.* EATCS Bulletin 28, 1986.

[25] Deepak Kapur, David R. Musser. *Proof by Consistency.* Artificial Intelligence 31 (1987), 125-157.

[26] Deepak Kapur, David R. Musser. *Inductive Reasoning with Incomplete Specifications.* Preliminary Report. IEEE symposium on Logic In Computer Science 1986.

[27] Deepak Kapur, Paliath Narendran, Hantao Zhang. *Automating Inductionless Induction.* J. Symbolic Computation (1991) **11**, pp. 83-111.

[28] Emmanuel Kounalis, Michaël Rusinowitch. *On Word Problems in Horn Theories.* 9^{th} International Conference on Automated Deduction 1988. LNAI 310. Springer-Verlag, Berlin 1988.

[29] Emmanuel Kounalis, Michaël Rusinowitch. *Mechanizing Inductive Reasoning.* Proc. of 8^{th} AAAI 1990, AAAI Press/The MIT Press, Menlo Park, Cambridge, London, Vol. 1, pp. 240-245.

[30] Peter Padawitz. *Horn Logic and Rewriting for Functional and Logic Program Design.* MIP-9002, März 1990, Fakultät für Mathematik und Informatik, Universität Passau, Postfach 2540, W-8390 Passau.

[31] David A. Plaisted. *Semantic Confluence Tests and Completion Methods.* Information and Control **65**, 1985, pp. 182-215.

[32] Uday S. Reddy. *Term Rewriting Induction.* Proc. of 10^{th} CADE 1990, LNAI 449. Springer-Verlag, Berlin 1990.

[33] Claus-Peter Wirth. *Inductive Theorem Proving in Theories specified by Positive/Negative Conditional Equations.* Diplomarbeit, 1991, Universität Kaiserslautern, Fachbereich Informatik, Postfach 3049, W-6750 Kaiserslautern.

[34] Hantao Zhang. *Reduction, Superposition and Induction: Automated Reasoning in an Equational Logic.* Rensselaer Polytech. Inst., Dept. of Comp. Sci., Troy, NY, PhD thesis, 1988.

[35] Hantao Zhang, Deepak Kapur. *First-Order Theorem Proving using Conditional Rewrite Rules.* 9^{th} International Conference on Automated Deduction 1988. LNAI 310. Springer-Verlag, Berlin 1988.

[36] Hantao Zhang, Deepak Kapur, Mukkai S. Krishnamoorthy. *A Mechanizable Induction Principle for Equational Specifications.* 9^{th} International Conference on Automated Deduction 1988. LNAI 310. Springer-Verlag, Berlin 1988.